

Мазурок Т. Л.

доктор технічних наук, професор кафедри прикладної математики та інформатики

Сметаніна Л. С.

кандидат педагогічних наук,

доцент кафедри прикладної математики та інформатики

Корабльов В. А.

старший викладач кафедри

прикладної математики та інформатики

ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ В КОМЕРЦІЙНІЙ ДІЯЛЬНОСТІ

Змістові лінії курсу:

Змістовий модуль 1. Основи інформаційних технологій

Тема 1.1. Поняття інформації. Особливості інформаційних технологій. Роль інформаційних технологій у комерційній діяльності.

Тема 1.2. Поняття, призначення та структура інформаційної системи.

Тема 1.3. Класифікація сучасних інформаційних технологій.

Тема 1.4. Підготовка текстових документів. Об'єкти та основні операції текстового редактора.

Змістовий модуль 2. Засоби електронної комерції

Тема 2.1. Технологія електронних таблиць.

Тема 2.2. Застосування ЕТ для прийняття рішень в комерційній діяльності.

Тема 2.3. Інтернет-технології. Створення веб-сайтів.

Тема 2.4. Засоби підготовки електронної презентації.

Змістовий модуль 3. Апаратне та системне програмне забезпечення персонального комп'ютера

Тема 3.1. Компоненти сучасного комп'ютера.

Тема 3.2. Операційні системи

Розглянемо приклади завдань з декількох тематичних розділів курсу та теоретичні аспекти.

Тема 1.4. Робота із таблицями у текстовому редакторі. Оформлення замовлення на обладнання.

Мета роботи – навчитись працювати з таблицями у текстовому редакторі з метою вирішення прикладних задач професійної діяльності. Для цього змодельємо замовлення вашої фірми на обладнання в новий офіс (або канцтовари) з урахуванням цін кожного найменування та вартості всього замовлення.

Розглянемо створення такого документа по кроках.

1. Відкрийте новий документ у редакторі Word. (Файл \ Створити \ Новий документ)

2. Сторінка замовлення буде розташовуватися не вертикально, а горизонтально. Щоб здійснити такий поворот листа зайдіть в меню Файл \ Параметри сторінки \ Розмір паперу. І виберіть альбомну орієнтацію.

3. Встановіть наступні параметри сторінки і шрифтів:

Орієнтація: альбомна.

Розмір паперу: А4 210x297мм.

Верхнє, нижнє, лівє і правє поля: 2см.

Палітурка: 0.

Масштаб відображення документа: по ширині сторінки (щоб бачити на екрані весь рядок прайс-листа).

4. Наберемо заголовок:



Замовлення на обладнання на 28.12.2014 р

Шрифт виберіть: Times New Roman, напівжирний, розмір 16 пунктів.

Форматування: по центру сторінки.

Збільшимо інтервал між буквами першого рядка так, щоб рядок став розріджений, для цього: Формат \ Шрифт \ Інтервал = 2 пункту

5. Перенесемо форматування першого рядка на другу (застосуємо ті ж властивості для тексту на другому рядку), для цього використовуємо кнопку «Формат за зразком» панелі інструментів «Стандартна». (Кнопка з пензликом) Для другого рядка тексту виберемо розмір шрифту 12 пунктів.

6. Координати вашого підприємства і фірми, до якої надійде замовлення, розмістимо в таблиці з двох горизонтально розташованих осередків, причому в лівій клітинці вирівнюємо рядки по лівому краю, а в правій клітинці – по правому краю.

7. Лінії обрамлення таблиці зробимо невидимими, крім верхнього краю таблиці. Тоді назва фірми вийде підкресленою.

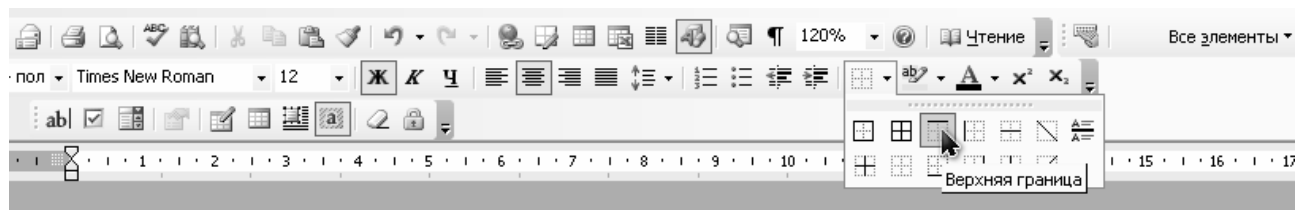
Для цього:

8. Встановимо курсор на початок наступного рядка і натиснемо кнопку Додати таблицю (або меню Таблиця \ Додати таблицю \ Набираємо число рядків і стовпців).

9. Виділимо в таблиці два осередки по горизонталі. (Виділена частина таблиці показує розмір таблиці, яка буде вставлена в документ).

10. Зовнішня і внутрішня границі нової таблиці за замовчуванням проведені суцільною лінією товщини 0,5 пункту. Викличте на екран панель інструментів Таблиці і межі. Оберіть товщину лінії 1,5 пункту.

11. Натисніть на кнопку «Зовнішні границі» і виберіть у списку зразок для скасування всіх кордонів в таблиці. Колір меж таблиці стане сірим. Хоча такі кордони і видно на екрані, на друк вони не виводяться.



Замовлення на обладнання на 28.12.2014 р



--	--

12. На панелі інструментів оберіть зразок «Верхня межа». З'явиться лінія вздовж верхньої межі таблиці.

13. У лівій комірці таблиці введіть координати вашої фірми-замовника.

14. Наприкінці перших двох рядків натискайте комбінації клавіш Shift + Enter для того, щоб у поточному абзаці переходити на новий рядок, але весь текст, що вводиться залишався б одним абзацом. Абзац в комірці таблиці вирівнюємо по лівому краю комірки.

15. Натисніть клавішу Tab, щоб перейти в праву половину таблиці і введіть аналогічно текст координат фірми, у якій ви будете купувати товар.

16. Відформатуйте цей текст по правому краю комірки. Повинно вийти, наприклад, наступне:

**Замовлення на обладнання
на 28.12.2014 р**

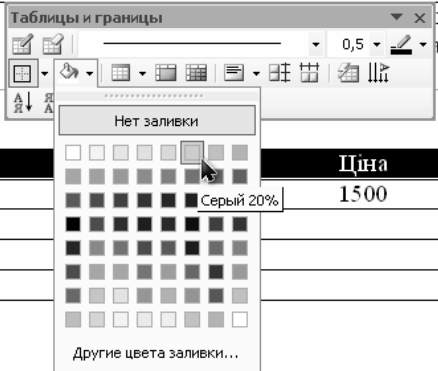
Мебельна фабрика «12 стільців» м. Одеса вул. Велика Арнаутська, б.133 тел. 048 7877788	Комерційна установа «О. Бендер» м. Одеса вул. Мала Арнаутська, б.33 тел. 048 3478788
--	--

17. Вставимо у кінець документа таблицю з чотирьох рядків і двох колонок для замовлення, в якій будемо писати найменування товару, що замовляється. (Меню Таблица \ Додати таблицю).

18. Встановимо необхідні лінії уздовж меж осередків таблиці і виділимо перший рядок яким-небудь кольором. Для виділення використовуйте кнопку «Колір заливки». Виберіть заливку, наприклад, сірим кольором 20%. В осередках першого рядка виконаємо вирівнювання по центру.

19. Введіть найменування товарів у першому стовпчику і відповідну їм ціну у другому. Наше замовлення складається з 10 пунктів. Приблизно виходить наступне.

**Замовлення на обладнання
на 28.12.2014 р**

Мебельна фабрика «12 стільців» м. Одеса вул. Велика Арнаутська, б.133 тел. 048 7877788	
Найменування	Ціна
Комп'ютерний стіл	1500

20. Вставте додаткові третю і четверту колонки (для цього виділіть другий стовець \ меню Таблица \ Додати \ Стовець праворуч)

21. У третій стовець введіть кількість товару, а четверту просто найменуйте – «Вартість».

22. Далі необхідно перерахувати вартість кожного товару в залежності від кількості. Ми зробимо це за формулою.

23. Але для того щоб користуватися формулами перерахунків, необхідно якимось позначати осередки. Прийнято позначати їх так, наприклад:

A1	B1	C1	D1
A2	B2	C2	D2
A3	B3	C3	D3
A4	B4	C4	D4

24. Тепер, знаючи як називається кожна клітинка, можемо звертатися до будь-якої. Зупинимось докладно на формулі.

Для того щоб знайти вартість, наприклад, всіх замовлених столів, необхідно ціну одного столу помножити на їх кількість. Тобто вміст комірки B2 помножити на вміст комірки C2. Виконаємо це за допомогою формули.

25. Встановіть текстовий курсор в комірку таблиці d2 і виберіть меню Таблиця \ Формула.

Якщо Ви працюєте у текстовому редакторі версії 2007 та вище, формули розташовані на ленті у розділі «Макет».

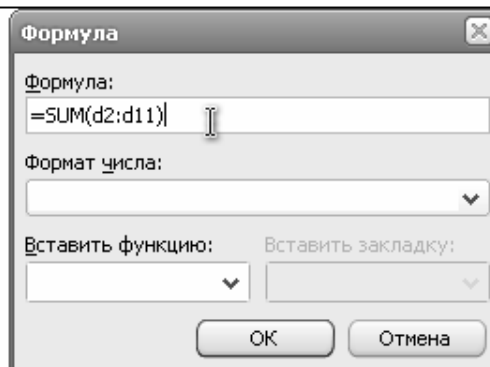
У полі введення «Формула» після знаку рівності введіть B2 * C2 і натисніть кнопку Ок. У комірці C2 з'явиться значення, що є твором чисел з комірок B2 і C2. Адреси комірок можна вводити і на нижньому регістрі, наприклад b2 * c5.

Аналогічні формули внесіть в усі осередки, що залишились.

26. Підрахуємо загальну вартість замовлення. Для цього додамо в кінець таблиці рядок (аналогічно вставці стовпчика).

27. У третьому стовпці останнього рядка введемо «Всього»: і в четвертому за формулою підрахуємо суму замовлення, скориставшись функцією SUM (). У дужках цієї функції вказується з якого осередку по який в четвертому стовпці буде проводитися підсумовування. Наприклад так:

5	3500
4	1200
7	800
Всього:	



28. Замовлення практично готове. Але виявилось, що другого найменування необхідно на 2 більше. Виходить, вам необхідно все перерахувати. Саме для цієї ситуації ми й користувалися формулами, щоб при внесенні змін за допомогою одного натиснення на кнопку здійснювалось перерахування даних всієї таблиці.

29. Щоб цього домогтися, змініть, наприклад число в комірці C3, збільшивши на 2. Виділіть останній стовпець, в який вводились формули і натисніть клавішу F9. Значення відповідних осередків будуть перераховані, а також перерахується і підсумкова вартість.

30. Збережіть отриманий документ у своїй папці. Захищаючи роботу Ви повинні будете вміти виконувати автоматичний перерахунок за проханням викладача.

Тема 2.1.Робота з електронними таблицями. Використання форм для звіту.

Кожен керівник підприємства стежить за динамікою роботи. Якщо це комерційна організація з декількох відділів, яка, наприклад, займається продажем, то дуже актуальною є необхідність наочного представлення звітної інформації по кожному відділу за певний період часу.

Завдання полягає в тому, щоб створити універсальний бланк-звіт, у якому можна вибрати будь-який відділ зі списку-форми з автоматичною побудовою динаміки змін його роботи.

Розглянемо етапи роботи.

1. Відкрийте програму MSExcel.

На першому аркуші наберіть Назви відділів та результат їх роботи за рік. Наприклад.

	A	B	C	D	E	F	G
1	Динаміка продажів у структурних підрозділах підприємства						
2							
3	Відділ	січень	лютий	березень	квітень	травень	червень
4	Південний	45000	67000	34000	14000	50000	35000
5	Центральний	17000	57000	47000	37000	23500	38900
6	Східний	34000	45000	67000	26000	48600	45300
7	Роздрібний	10000	25000	16700	4500	25000	12800
8							
9							

Назвіть перший лист "Відділення".

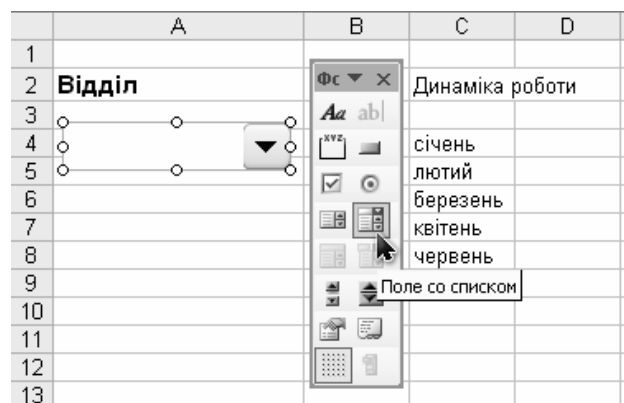
Другий лист назовемо "Бланк". На ньому буде розташований графік, побудований за результатами продажів протягом року, самі результати, а також форма з переліком відділів.

	A	B	C	D	E
1					
2	Відділ			Динаміка роботи	
3					
4				січень	
5				лютий	
6				березень	
7				квітень	
8				червень	

Під прізвищем розмістимо форму – «Поле зі списком». Для цього – меню Вид \ Панелі інструментів \ Форми. На панелі, що з'явилася виберіть інструмент «Поле зі списком».

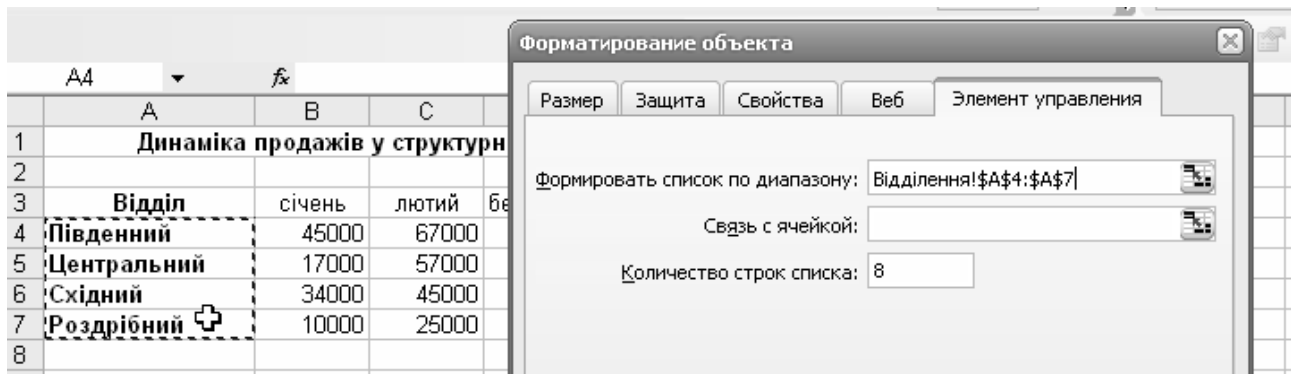
Додайте його на аркуш і надайте необхідний розмір.

Якщо Ви працюєте у табличному редакторі версії 2007 та вище, форми розташовані на ленті у розділі «Розробник». Якщо цього розділу не має на ленті – виведіть його за допомогою налаштувань.

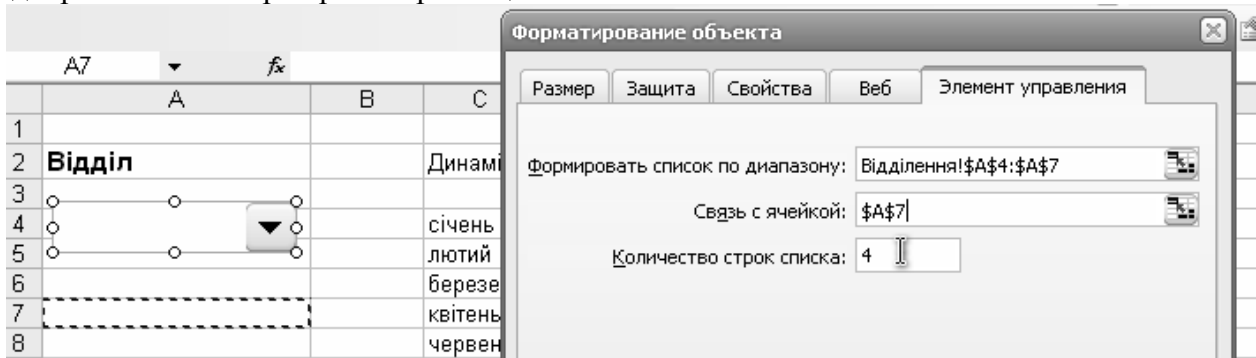


Тепер необхідно наповнити наш список наявними на першому аркуші назвами відділів. Для цього - Права кнопка миші \ Формат об'єкта.

Формувати список з – перейдіть на Лист1 і виділіть всі назви відділів.



Зв'язати з осередком – виберіть будь-яку клітинку на другому аркуші в якій буде відображатися номер обраної прізвища. Натисніть ОК.

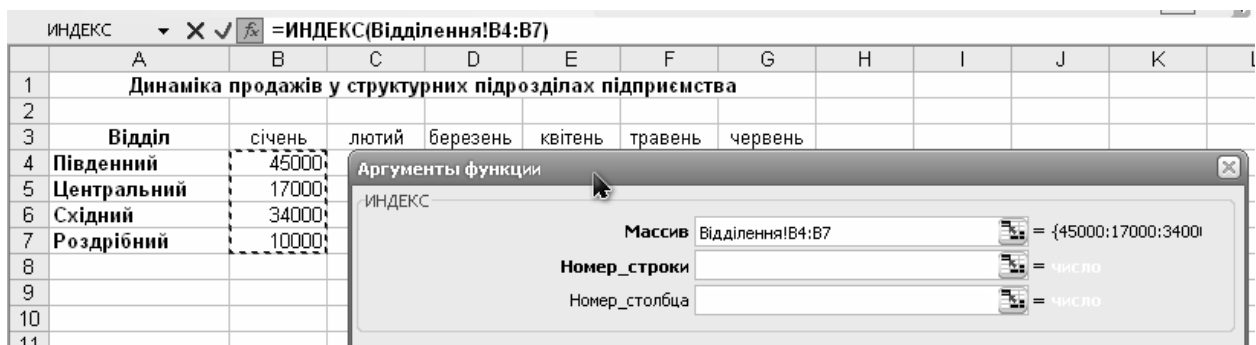


У стовпці С вкажемо номера назви місяців за які буде наведений звіт. У нашому прикладі це місяці з січня по червень, у Вашому – з січня по грудень.

У стовпці D навпроти кожного місяця повинен виводитися результат продажу для обраного у полі зі списком відділі. Для цього використовуємо функцію

ИНДЕКС (масив з якого формуємо список; № осередку по якому виводимо запис.)

У нашому прикладі «Масив» - це усі суми з чотирьох відділень за перший місяць. Необхідно поставити курсор у осередок «Масив», перейти на перший лист «Відділення» та виділити перший стовпчик.



Переходимо у «Номер осередку» та обираємо осередок, з яким зв'язували форму «Поле зі списком». У нашому випадку це A7.

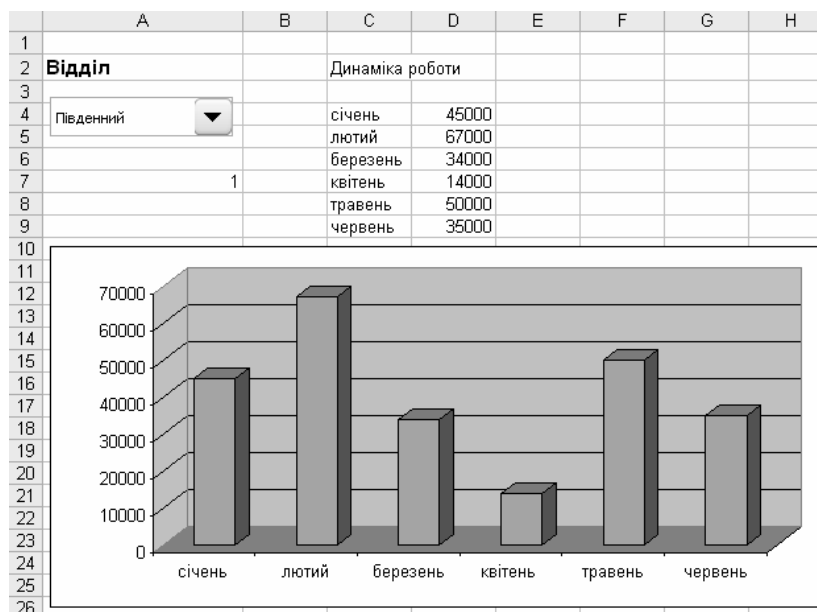
Аналогічно для наступних осередків.

Тепер побудуємо графік за отриманим даним. Для цього виділимо стовпець D, Меню Вставка \ Діаграма \ оберемо тип графіку.

Далі самостійно уважно читаємо кожен крок майстра і натискаємо Далі.

Подивіться як змінюється графік після вибору різних відділів.

Результат приблизно наступний.



Тема 3.1. Компоненти сучасного комп'ютера

Розглянемо тлумачення основних термінів і понять.

Плата – плоска текстолітова основа, частіше прямокутної форми, на якій кріпляться й з'єднуються між собою електричні компоненти. Часто є можливість стикувати плату з іншими компонентами комп'ютера. Кожна конкретна плата виконує свої, визначені функції: вона може бути відповідальна, наприклад, за виведення інформації на монітор, або за виведення звуку через колонки, або забезпечувати зв'язок Вашого комп'ютера з іншими комп'ютерами за допомогою спеціального кабелю або наявних телефонних комунікацій, загалом, кожна конкретна плата в комп'ютері виконує якісь властиві їй функції.

Чіп – синонім слова мікросхема. Колись найпростішим елементом електричного логічного пристрою була електронна лампа, потім транзистор, сьогодні ступінь інтеграції електричних компонентів настільки висока, що в одній мікросхемі (чіпі) розташовуються десятки мільйонів транзисторів. У чіпі електричне реалізується якась логічна схема, яка і дозволяє чипу виконувати ті функції, для яких він призначений. Про функції чіпа: чіп може, наприклад, являти собою фізичне місце зберігання інформації (бути, як кажуть, чіпом пам'яті), або вміти виконувати якісь розрахунки, або дозволяти сполучати певні компоненти один з одним. Наприклад - це чіпи пам'яті, чіп управління накопичувачами на жорстких дисках, чіп управління периферією і внутрішніми пристроями (на всьому цьому ми ще зупинимося).

Контролер – контролером прийнято називати деякий пристрій, що служить для керування іншим пристроєм (наприклад, жорстким диском) і для підтримки обміну даними з цим пристроєм. Наприклад: Вам необхідно підключити до комп'ютера принтер: значить, в комп'ютері повинен бути спеціальний контролер, який забезпечить можливість функціонування Вашого комп'ютера разом з принтером. Що прийнято називати контролером: контролером називають плату, яка забезпечує підключення пристроїв, також контролером називають чіп, припаяний до материнської плати, і вставляти додаткову плату немає необхідності, іноді контролером називають частину чіпа - якщо один чіп є контролером відразу багатьох пристроїв. При цьому в явному вигляді ніяку частину чіпа природно не окреслюють кружком і не кажуть що тут один контролер, а тут інший, немає, просто кажуть, що в даному чіпі реалізовано кілька контролерів.

Порт – термін порт вживається в декількох значеннях, нас же зараз цікавить наступне: порт - роз'єм для підключення пристроїв до контролера. У вже наведеному прикладі з принтером в системі повинен бути контролер принтера і повинен бути роз'єм

(порт) для підключення цього принтера. Тому, до речі, зазвичай говорять не контролер принтера, а контролер порту принтера.

Шина – це магістраль, що зв'язує деякі компоненти комп'ютера між собою. Фізично, шина - це провідники, що забезпечують передачу електричних сигналів між пристроями. Характеризується шина швидкістю, з якою по ній можуть передаватися дані, а також кількістю провідників, по яких можуть одночасно передаватися дані.

Інтерфейс – дуже широке поняття, воно включає в себе засоби і способи взаємодії різних об'єктів між собою. Ви, мабуть, знаєте, що інтерфейсом називають, наприклад, зовнішній вигляд програми, з якою ви працюєте, її вікна і меню, її елементи управління і т.д. У світі Hardware інтерфейсом називають засоби взаємодії між різними пристроями. У поняття інтерфейсу включається як порт, так і контролер, який служить, наприклад, для підключення цього принтера, в цьому випадку мова йде про інтерфейс підключення принтера.

Комп'ютером ми будемо називати вміст системного блоку, корпусу, в якому зосереджені всі компоненти відповідальні за обчислення, зберігання, обробку та передачу даних. Власне це і є комп'ютер. Все, що знаходиться зовні - пристрої, що служать для введення або виведення інформації. Що у нас знаходиться зовні системного блоку? Монітор, принтер - пристрої виведення інформації на екран і папір відповідно, клавіатура, миша - пристрої введення інформації і. Всі ці пристрої не служать для зберігання, обробки даних, в них не проводяться обчислення. Це так звані периферійні пристрої. В принципі комп'ютер може працювати і без них. Навпаки - в системному блоці знаходяться компоненти, які виробляють обчислення, в яких зберігаються і обробляються дані.

Центром комп'ютера є найбільша плата - так звана материнська плата (Mother Board). Роль цієї плати вкрай важлива: вона є як би сполучною ланкою між усіма компонентами комп'ютера, практично всі пристрої підключаються саме до материнської плати. Природно від можливостей материнської плати багато в чому залежать можливості комп'ютера.

В першу чергу слід звернути увагу на процесор (CPU, Central Processor Unit, Модуль центрального процесора).

Процесор – це, звичайно, найбільший чіп в комп'ютері. Процесор це пристрій, який виконує певний набір команд (інструкцій), а комп'ютерна програма - це і є послідовність цих самих інструкцій. Тобто процесор - як би мозок комп'ютера, процесор виконує практично всі обчислення в комп'ютері. Природно, що від швидкості виконання інструкцій (продуктивності процесора) безпосередньо залежить і сумарна продуктивність всього комп'ютера.

Наступний найважливіший компонент, що підключається до материнської плати оперативна пам'ять (RAM, Random Access Memory, пам'ять з довільним доступом). Оперативна пам'ять є робочою областю для процесора, в ній розміщуються програми під час їх виконання процесором, і дані які обробляють ці програми. Оперативна пам'ять є енергозалежним пристроєм, тобто дані в ній зберігаються тільки при включеному живленні комп'ютера. Коли комп'ютер вимикається, всі її вміст стирається. Всі данні попередньо зберігаються на жорсткому диску, який є пристроєм для тривалого зберігання інформації (навіть при відсутності живлення). Оперативна пам'ять приблизно в сотню разів (при рівному обсязі зберігання даних) дорожче жорсткого диска. Тому в системі встановлюють деяку кількість оперативної пам'яті, яке набагато менше, ніж обсяг сучасного жорсткого диска.

Таким чином, дані і програми одного разу зчитуються з жорсткого диска і розміщуються в оперативній пам'яті, а потім робота процесора з цими програмами і даними відбувається в оперативній пам'яті. Природно, що від швидкодії оперативної пам'яті залежить, наскільки швидко процесор забезпечується даними для обробки і продуктивність комп'ютера в цілому. Наприклад, комп'ютер з більш швидким процесором може конкретну задачу вирішити повільніше, ніж комп'ютер з менш швидким процесором, якщо в останнього, оперативна пам'ять швидше. Важливо підкреслити, що тут не можна говорити про те, що один комп'ютер швидше іншого, потрібно розуміти, що це залежить від кожної

конкретної задачі. Так само важливим є питання про те, як залежить продуктивність комп'ютера від кількості встановленої оперативної пам'яті. Не можна говорити про те, що чим більше пам'яті, тим швидше комп'ютер, все залежить від конкретного завдання, яке вирішує комп'ютер.

Наступний найважливіший компонент, на якому ми зупинимося - так званий набір мікросхем або chipset, на базі якого будується материнська плата. Чіпсет - самі великі (після процесора) мікросхеми в комп'ютері, і найбільші з припаяних до материнської плати. Чіпсет забезпечує зв'язок між основними вузлами, розташованими на материнській платі, в першу чергу між процесором і пам'яттю. Тому, природно, від чіпсета так само залежить продуктивність комп'ютера в цілому, тому якщо, наприклад, чіпсет повільно працює з пам'яттю, то і система працює повільніше, ніж система з тим же процесором і пам'яттю, але іншим, більш швидко працюють з пам'яттю чіпсетом.

Але функція зв'язування всіх компонентів в єдину систему не єдина функція чіпсета. Крім того, сучасний чіпсет містить цілий ряд основних, базових контролерів різних пристроїв, що підключаються до материнської плати. Навіщо купувати для підключення дисководу окрему плату контролера дисководу, якщо контролер можна інтегрувати в чіпсет. Зрозуміло, собівартість інтеграції контролера на материнську плату набагато нижче придбання окремої плати контролера.

Тому багато контролерів інтегрують на плату (в даному випадку в чіпсет).

Контролер дисководу (його називають FDC - Floppy Disk Controller, а сам дисковод FDD - Floppy Disk Drive), до нього можна підключити 2 дисководу.

Контролер жорсткого диска. Їх може бути кілька різновидів. Parallel ATA Controller (PATA) або IDE Controller, причому вбудований в чіпсет контролер підтримує 2 порти для підключення жорстких дисків, а до кожного порту можна підключити по 2 диска, тобто можна підключити до 4 жорстких дисків. Serial ATA Controller (SATA), новий інтерфейс для підключення жорстких дисків, що володіє більш високою швидкістю передачі даних, і приєднанням тільки одного пристрою до порту (хоча портів може бути багато).

Контролер порту принтера (ще його називають паралельний порт або LPT port). Як зрозуміло з назви до цього порту підключають принтер, так само раніше в цей порт нерідко підключали сканер.

Комунікаційні порти (звичайно 2 шт.), Говорять так само про послідовних портах, COM - портах. До цих портів може підключатися миша, модем (пристрій для зв'язку з іншими комп'ютерами по телефонних лініях), деякі екзотичні принтери (зазвичай від мобільних комп'ютерів) і т.д.

Контролер клавіатури і контролер порту миші. Порти цього контролера називається PS / 2. Тому говорять про PS / 2 порт миші і клавіатури. Причому крім миші і клавіатури до них нічого більше не можна підключити.

Контролери універсальної послідовної Шини (USB, Universal Serial Bus). Ця шина розроблена в якості універсальної (що зазначено в назві) для підключення периферії. За допомогою цього контролера до комп'ютера можна підключити клавіатуру і мишу, принтер і сканер, модем і цифрову камеру, зовнішній носій інформації і навіть інший комп'ютер, тобто практично всі пристрої. Зручність і простота використання USB веде до того що він поступово замінює всі інші комунікаційні порти - вже зараз можна зустріти материнські плати, на задній панелі яких немає COM і LPT портів, зате є до шести USB (а найновіші чіпсети підтримують десять портів).

Мережевий контролер, LAN. Це пристрій дозволить комп'ютеру підключатися до локальної мережі, і обмінюватися даними з іншими комп'ютерами.

Часто низько швидкісні контролери (LPT, COM, PS / 2, FDC) встановлюють у вигляді додаткової мікросхеми, SIO - super input / output, для того, щоб винести з південного моста компоненти, продуктивність яких занадто мала.

Крім того, в чіпсети іноді інтегрують і інші, наприклад відео-контролер і контролер послідовної шини FireWire (IEEE 1394). Це ще одна високошвидкісна, що стає все

популярнішим послідовна універсальна шина. Її основним призначенням було підключення до ПК цифрових відеокамер, сьогодні спектр пристроїв розширено зовнішніми накопичувачами, оптичними приводами та іншими пристроями.

Також важливим компонентом материнської плати є мікросхема BIOS (Basic Input Output-System, базова система введення-виведення). Ця мікросхема є пам'ять, але не оперативну, а, навпаки, постійну. У цій мікросхемі записана програма, яка забезпечує початковий старт комп'ютера.

Ця програма і називається BIOS. (На ім'я цієї програми і саму мікросхему також іноді називають BIOS). Справа в тому, що комп'ютер, взагалі кажучи, лише купа заліза і пластику. І змусити цю купу що-небудь робити може лише програмне забезпечення. І в цій мікросхемі якраз записана програма, яка забезпечує початковий старт комп'ютера.

У момент старту комп'ютера в першу чергу починає виконуватися вміст мікросхеми постійної пам'яті, власне програма BIOS. Ця програма забезпечує процедуру старту машини, перевірки, ініціалізації і налаштування всіх її вузлів, а потім передає управління операційній системі.

При початковому старті комп'ютера і початкової ініціалізації пристроїв є цілий ряд параметрів, якими BIOS необхідно користуватися. Найбільш інтуїтивно зрозумілі параметри: системний час і дата. Адже користувач не вводить їх самостійно при кожному включенні комп'ютера (хоча на старих машинах так бувало). Крім цих є безліч інших параметрів, необхідних для початкового старту машини і які користувач може змінювати. Де ж зберігаються значення цих параметрів? Природно, в спеціально відведеній для цих цілей пам'яті, званої CMOS. Мікросхема пам'яті CMOS має ємність 2-4 кб і розташовується на материнській платі. Ця пам'ять не є енергонезалежною і харчується (спільно з годинником) від розташованої на платі батарейки.

Відеокарта - плата, що відповідає за формування і виведення зображення на монітор або інший пристрій відображення відеоінформації. Її основна мета - це візуалізувати роботу з комп'ютером, а в сучасних умовах ще й допомогти комфортно відпочити, проходячи рівні улюбленої гри. Відеокарта може бути інтегрована на материнську плату, найчастіше так чинять в недорогих офісних комп'ютерах, що вимагають від неї виконувати тільки своє основне завдання - виведення зображення на монітор. От тільки сучасні ігри вимагають від відеокарти більше, тому на ній присутній свій спеціалізований процесор (відео процесор, GPU) і своя швидка пам'ять (відео пам'ять), що перетворюють цю плату в 3D-прискорювач.

Зовнішня відео карта, тобто виконана у вигляді окремої плати, встановлюється в спеціальний роз'єм на материнській платі (втім як і інші плати розширення) - слот, частина деякої шини використовувану для роботи зі знімними пристроями.

У червні 1992 року Intel розробляє нову шину для підключення периферійних пристроїв - PCI (Peripheral Component Interconnect). Ця шина на відміну від VL-Bus є процесоронезалежною. Ця шина має розрядність 32 біта, працює на частоті 33 МГц, за безпечуючи, таким чином, пропускну спроможність 133 Мбайт / с (Спробуйте перерахувати самі. Вийшло? Тільки врахуйте що реальна частота цієї шини 33,333 ... МГц, тому що виходить з опорної 100 МГц шляхом ділення на 3).

Шина AGP має ширину 32 біта (як і PCI, власне AGP це і є модифікована шина PCI), але працює на частоті 66 МГц, маючи, таким чином, вдвічі більшу пропускну здатність (266 Мбайт / с). Але цим справа не обмежується. Справа в тому, що шина AGP має кілька режимів роботи. При чому у всіх цих режимах частота роботи шини однакова і становить 66 МГц. Але пропускну здатність шини AGP в різних режимах відрізняється. У режимі роботи, який називають AGP 2x, пропускну здатність шини становить 533 Мбайт / с.

PCI-Express, нова шина, призначена для заміни всіх нині існуючих (в першу чергу PCI і AGP). Шина PCI-Express є послідовною (т.е. інформація передається тільки по одному провіднику), двобічної (тобто для передачі даних в кожному напрямку використовується окремий провідник) високочастотної шиною (працює на частоті 2500 МГц). Для забезпечення стабільності на таких високих частотах дані розбиваються на пакети (частини) і кожен пакет доповнюється контрольною сумою, на які потрібно 20% пропускну

спроможності (про методи кодування і забезпечення цілісності переданих даних ми ще поговоримо в цьому курсі і до курсу Мережі). Не важко підрахувати, що пропускна здатність цієї шини становить $1 \times (2500 - 20\%) / 8 = 250 \text{ Мб / с}$ у кожному напрямку, тобто майже в 2 рази більше ніж у звичайної PCI. До того ж тепер пристрою не ділять пропускну здатність шини між собою - нова шина забезпечує кожний слот індивідуальним каналом.

Основні компоненти системного блоку ми вже розглянули. Тепер, перейдемо до периферійних пристроїв, які оточують системний блок, і дозволяють користувачеві взаємодіяти з комп'ютером. Периферійні пристрої можна розділити на наступні групи:

- пристрої введення інформації;
- пристрої виведення інформації;
- пристрої зберігання інформації;
- мультимедійні пристрої;
- пристрої передачі інформації.

Тема 3.2. Операційні системи

Операційна система найбільшою мірою визначає вигляд всієї обчислювальної системи в цілому. Незважаючи на це, користувачі, активно використовують обчислювальну техніку, часто зазнають труднощів при спробі дати визначення операційній системі. Частково це пов'язано з тим, що ОС виконує дві по суті мало пов'язані функції:

- забезпечення користувачу-програмісту зручностей за допомогою надання для нього розширеної машини;
- підвищення ефективності використання комп'ютера шляхом раціонального управління його ресурсами.

Програма, яка приховує від програміста всі реалії апаратури і надає можливість простого, зручного перегляду зазначених файлів, читання чи запису - це, звичайно, операційна система.

Функцією ОС є надання користувачу деякої розширеної або віртуальної машини, яку легше програмувати і з якою легше працювати, ніж безпосередньо з апаратурою, складовою реальну машину.

Згідно з другим підходом функцією ОС є розподіл процесорів, пам'яті, пристроїв і даних між процесами, конкуруючими за ці ресурси. ОС повинна управляти всіма ресурсами обчислювальної машини таким чином, щоб забезпечити максимальну ефективність її функціонування. Критерієм ефективності може бути, наприклад, пропускна здатність або реактивність системи.

Управління ресурсами включає рішення двох загальних, що не залежать від типу ресурсу завдань:

1. планування ресурсу - тобто визначення, кому, коли, а для подільних ресурсів і в якій кількості, необхідно виділити даний ресурс;
2. відстеження стану ресурсу - тобто підтримка оперативної інформації про те, зайнятий або не зайнятий ресурс, а для подільних ресурсів - яка кількість ресурсу вже розподілено, а яке вільно.

Операційна система Linux

Коріння Linux простежуються ще з 70-х років 20-го століття. Точкою відліку можна вважати появу операційної системи Unix в 1969-му році в США у фірмі Bell Laboratories, дочірньому підрозділі компанії AT & T. Unix стала основною для великої кількості операційних систем промислового класу.

Linux ж найбільше зобов'язаний своїм життям двом проектам - GNU і Minix.

GNU

Історія проекту GNU почалася у вересні 1983-го року. Основоположник проекту GNU, Річард Столлман (Richard M. Stallman) працював у цей час в лабораторії штучного інтелекту Массачусетського технологічного інституту (Massachusetts Institute of Technology,

MIT, Cambridge, Massachusetts). Столлман називають одним з найвидатніших програмістів нашого часу.

У тому середовищі, до якої належав Столлман, було прийнято вільно обмінюватися програмами і їх початковими кодами. Ліцензія ж на Unix від AT & T, наприклад, коштувала 40 000 доларів. Дозволити собі купити її могли тільки досить великі фірми. А без володіння ліцензією, програміст не мав права використовувати вихідні коди системи в своїх розробках. Це перешкоджало обміну ідеями в сфері програмування і сильно гальмувало процес створення програм, оскільки замість того, щоб запозичити готовий шматок коду для вирішення тієї чи іншої задачі, розробник програми був змушений писати цю частину коду заново, що те саме винаходу колеса.

Абревіатура GNU розшифровується як «GNU - це не Unix» (GNU is Not Unix). Unix завжди була закритим програмним, тобто вона позбавляє своїх користувачів свободи співробітництва, а також контролю над своїми комп'ютерами (як Windows в наші дні). Трохи пізніше Столлман написав свій знаменитий Маніфест GNU, який став основою для ліцензії GPL (GNU General Public License). Роль цієї ліцензії можна переоцінити, вона змінила всю комп'ютерну індустрію.

Основна ідея GPL полягає в тому, що користувач повинен мати такими чотирма правами (або чотирма свободами):

- Правом запускати програму для будь-яких цілей (свобода 0);
- Правом вивчати пристрій програми і пристосовувати її до своїх потреб (свобода 1), що передбачає доступ до вихідного коду програми;
- Правом поширювати програму, маючи можливість допомогти іншим (свобода 2);
- Правом покращувати програму і публікувати поліпшення, на користь всієї спільноти (свобода 3), що теж передбачає доступ до вихідного коду програми.

Програмне забезпечення, яке розповсюджується під цією ліцензією, можна як завгодно використовувати, копіювати, допрацьовувати, модифікувати, передавати або продавати модифіковані (або навіть не модифіковані) версії іншим особам за умови, що результат такої переробки теж буде поширюватися під ліцензією GPL. Остання умова - найважливіше і визначальне в цієї ліцензії. Воно гарантує, що результати зусиль розробників вільного ПЗ залишаться відкритими і не стануть частиною якогось ліцензованого звичайним способом продукту. Воно також відрізняє вільне ПЗ від ПО, розповсюдженого безкоштовно. Одна з вимог цієї ліцензії полягає в тому, що продаючи ПЗ під ліцензією GPL, ви повинні надати вихідні коди цього ПЗ кожному, хто захоче отримати до них доступ. Ліцензія GPL «робить ПО вільним і гарантує, що воно залишиться вільним».

До 1990-го року в рамках проекту GNU було створено більшість компонент, необхідних для функціонування вільної операційної системи. Крім текстового редактора Emacs, Столлман створив компілятор gcc (GNU C Compiler) і відладчик gdb. Будучи видатним програмістом, Річард Столлман поодиноці зумів створити ефективний і надійний компілятор, який перевершує за своїми якостями продукти комерційних постачальників, створювані цілими групами програмістів. Оскільки спочатку при його створенні ставилося завдання забезпечення переносимості, сьогодні існують версії цього компілятора практично для всіх операційних систем. Пізніше були створені компілятори і для інших мов програмування, включаючи C ++, Pascal і Fortran. Тому зараз абревіатура GCC розшифровується як GNU Compiler Collection.

Minix

Оскільки апаратні можливості персоналок стрімко зростали, природно, що рано чи пізно повинні були з'явитися варіанти Unix для комп'ютерів на основі Intel-сумісних процесорів.

Один з таких варіантів Unix-подібної операційної системи, який відіграв особливу роль в історії Linux, був розроблений в січні 1987-го року Ендрю Таненбаум (Andrew S. Tanenbaum), професором Університету Вріє, Амстердам, Нідерланди. Таненбаум був одним з провідних фахівців в області розробки операційних систем. Свою операційну систему Minix

(Мінікс) він розробив як навчальний посібник, на прикладі якого показував студентам внутрішній устрій реальної операційної системи.

Minix не була вершиною досконалості. Вона була орієнтована на мікропроцесор Intel 80286, який на той час панував на ринку. Але у неї було одне дуже важлива якість - відкриті вихідні коди. Кожен, хто мав книгу Таненбаума «Операційні системи», міг вивчити і проаналізувати 12 000 рядків коду, написаного на мові Сі і на асемблері. Це був той рідкісний випадок, коли вихідні коди були замкнені під сімома печатками в сейфах розробника. Чудовий автор, Таненбаум зумів залучити найвидатніші уми комп'ютерної науки в обговорення мистецтва створення операційних систем. Minix можна було придбати і окремо від книги, вона могла бути реально встановлена на персональний комп'ютер. Студенти комп'ютерних факультетів по всьому світу сиділи над книгою Таненбаума, вчитуючись в коди з метою зрозуміти, як працює та сама система, яка управляє їх комп'ютером. І одним з таких студентів був Лінус Торвальдс.

Linux

У 1991-му році, Лінус Торвальдс, фінський студент, надзвичайно захопився ідеєю написати сумісний з Unix ядро операційної системи для свого персонального комп'ютера з процесором Intel. Прототипом для майбутнього ядра стала операційна система Minix: сумісна з Unix операційна система для персональних комп'ютерів, яка завантажувалася з дискети і вміщувалася в дуже обмеженою в ті часи пам'яті персонального комп'ютера.

Той факт, що Лінус виклав код своєї ОС в Інтернет, був вирішальним у подальшій долі Linux. Хоча в 1991-му році Інтернет ще не був так широко поширений, як у наші дні, зате користувалися ним в основному люди, які мають достатню технічну підготовку. І вже з самого початку Торвальдс отримав кілька зацікавлених відгуків.

Спочатку до розробки долучилися сотні, потім тисячі, потім сотні тисяч добровільних помічників. Система вже не була просто іграшкою для хакерів. Доповнена масою програм, розроблених в рамках проекту GNU, ОС Linux стала придатна для практичного використання. А те, що ядро системи поширювалося під ліцензією GNU General Public License, гарантувало, що вихідні коди системи залишаться вільними, тобто зможуть копіюватися, вивчатися і модифікуватися без побоювання нарватися на якесь переслідування з боку розробника або якоїсь комерційної фірми. Цей факт привертав до лав користувачів і прихильників Linux все нових послідовників, в першу чергу з числа студентів і програмістів.

До цього часу сформувалася окрема конференція в Інтернеті, присвячена Linux, - comp.os.linux. Ентузіасти утворили безліч груп користувачів і на початку 1994-го року вийшов перший номер журналу «Linux Journal». Linux привернула увагу промислових фірм і кілька невеликих компаній почали розробляти і продавати власні версії Linux.

З технічної точки зору, Linux являє собою тільки ядро Unix-подібної операційної системи, що відповідає за взаємодію з апаратною частиною комп'ютера і виконання таких завдань, як розподіл пам'яті, виділення процесорного часу різними програмами і так далі. Крім ядра, операційна система включає в себе безліч різних утиліт, які служать для організації взаємодії користувача з системою. Успіх Linux як операційної системи багато в чому обумовлений тим, що до 1991-го року в рамках проекту GNU вже було розроблено безліч утиліт, вільно розповсюджуваних в Інтернеті. Проекту GNU не вистачало ядра, а ядро, швидше за все, залишилося б незатребуваним, якби були відсутні необхідні для роботи утиліти. Лінус Торвальдс опинився зі своєю розробкою в потрібному місці в потрібний час. І Річард Столлман прав, коли наполягає на тому, що операційну систему слід називати не Linux, а GNU / Linux. Але назва Linux історично закріпилося за цією ОС, тому ми теж будемо називати її просто Linux (не забуваючи про заслуги Столлман і його сподвижників).

Архітектура Linux

Linux організована у вигляді набору відносно незалежних блоків, які називаються завантажуються модулями (loadable modules). Завантажуванні модулі Linux мають дві відмінні риси.

Динамічне зв'язування. Будь модуль ядра може бути завантажений в пам'ять і приєднаний до ядра в той час, коли саме ядро вже знаходиться в пам'яті і виконується. Будь модуль може бути також від'єднаний від ядра і видалений з пам'яті в будь-який момент часу.

Стекова організація. Модулі організовані у вигляді певної ієрархічної структури. Окремі модулі можуть виконувати роль бібліотек при зверненні до них модулів більш високих рівнів в рамках цієї структури; вони самі також можуть звертатися до модулів на більш низьких рівнях.

Динамічне зв'язування полегшує налаштування системи і економить пам'ять, яку займає ядро. В системі Linux програма користувача або сам користувач може завантажувати або вивантажувати модулі за допомогою команд `insmod` і `rmmod`. Саме ядро керує роботою окремих функцій і в міру потреби завантажує потрібні модулі або вивантажує ті, потреба в яких вже відпала.

Крім того, стекова організація дозволяє задавати залежності модулів, що дає дві основні переваги.

1. Код, що є загальним для набору однотипних модулів (наприклад, драйвери схожих пристроїв), можна помістити в один модуль, що дозволяє скоротити кількість повторень.

2. Ядро може перевірити наявність в пам'яті потрібних модулів, утримуючись від вивантаження модуля, який потрібен для роботи інших, залежних від нього, і завантажуючи разом з новим необхідним модулем всі необхідні додаткові модулі.

5. Структура дистрибутива Linux

1. Ядро, модулі ядра, вихідний код

2. Системні програми, консоль, утиліти, компілятори, редактори і т.д. (Shell Bash, gcc, vim, nano

3.1 Сервер:

- samba - файловий сервер;
- apache - веб сервер;
- LAMP - apache + php + mySQL
- DNS - сервер доменних імен
- postfix, sandmail - поштовий сервер
- ProFTPD - ftp сервер

3.2 Графічний сервер XSERVER

- XORG - новий графічний сервер
 - xf86free - старий графічний сервер
5. Desktop - KDE, GNOME, XFCE і т.д.

6. Програми з підтримкою графічного інтерфейсу.

Файлова система Linux

Структуру файлової системи можна представити наочно у вигляді дерева, "коренем" якого є кореневий каталог, а у вершинах розташовані всі інші каталоги. На рис. зображено дерево каталогів, курсивом позначені імена файлів, прямим накресленням - імена каталогів.

/ bin Назва цього каталогу походить від слова "binaries" ("двоїчні", "виконувані"). У цьому каталозі знаходяться виконувані файли найнеобхідніших утиліт. Сюди потрапляють такі програми, які можуть знадобитися системного адміністратора або іншим користувачам для усунення неполадок в системі або при відновленні після збою.

/ boot "Boot" - завантаження системи. У цьому каталозі знаходяться файли, необхідні для самого першого етапу - завантаження ядра - і, звичайно, саме ядро. Користувачеві практично ніколи не потрібно безпосередньо працювати з цими файлами.

/ dev У цьому каталозі знаходяться всі наявні в системі файли-дірки: файли особливого типу, призначені для звернення до різних системних ресурсів і пристроїв (англ. "devices" - "пристрою", звідси і скорочена назва каталогу).

/ etc Каталог для системних конфігураційних файлів. Тут зберігається інформація про специфічні налаштування даної системи: інформація про зареєстрованих користувачів, доступних ресурсах, налаштуваннях різних програм.

/ home Тут розташовані каталоги, що належать користувачам системи - домашні каталоги, звідси і назва "home". Відділення усіх файлів, що створюються користувачами, від інших системних файлів дає очевидну перевагу: серйозне пошкодження системи або необхідність оновлення не торкнеться найбільш цінної інформації - користувача файлів.

/ sbin Каталог для найважливіших системних утиліт (назва каталогу - скорочення від "system binaries"): на додаток до утиліт / bin тут знаходяться програми, необхідні для завантаження, резервного копіювання, відновлення системи. Повноваження на виконання цих програм є тільки у системного адміністратора.

/ tmp Цей каталог призначення для тимчасових файлів: в таких файлах програми зберігають необхідні для роботи проміжні дані. Зазвичай каталог / tmp очищається при завантаженні системи.

/ usr Каталог / usr- це "держава в державі". Тут можна знайти такі ж підкаталоги bin, etc, lib, sbin, як і в кореневому каталозі. Однак в кореневий каталог потрапляють тільки утиліти, необхідні для завантаження і відновлення системи в аварійній ситуації - всі інші програми і дані розташовуються в підкаталогах / usr. Прикладних програм в сучасних системах зазвичай встановлено дуже багато, тому цей розділ файлової системи може бути дуже великим.

/ var Назва цього каталогу - скорочення від "variable" ("змінні" данні). Тут розміщуються ті данні, які створюються в процесі роботи різними програмами і призначені для передачі іншим програмам і системам (черги друку, електронної пошти та ін.) Або для відомості системного адміністратора (системні журнали, що містять протоколи роботи системи).

Список використаної та рекомендованої літератури

1. Ананьев О. М. Информационные системы и технологии в менеджменте : Текст лекций. Ч. 1. Основы сетевых технологий / О. М. Ананьев; Укоопспілка, Львів. комерц. акад. – Л., 2003. – 51 с.
2. Браткевич В.В. Информатика. Комп'ютерна техніка. Комп'ютерні технології: підручник для студ. вищих навч. закладів / В.В. Браткевич, М.В. Бутов, І.О. Золотарьова та ін.; За ред. О. І. Пушкаря. – К.: ВЦ Академія, 2002. – 704 с.
3. Дибкова Л.М. Информатика та комп'ютерна техніка: посіб. для студ. вищих навч. закладів/ Л.М. Дибкова.– К.: ВЦ Академія, 2002. – 320 с.
4. Глинський Я.М. Практикум з інформатики: навч. посіб. / Я.М. Глинський. – Львів: Деол, 1998. – 167с.
5. Глинський Я.М. Информатика: навч. посіб. Книжка 2. Інформаційні технології / Я.М. Глинський. – Львів: Деол, 2002. – 254 с.
6. Шестопапов Є.А. Информатика. Базовий курс. / Є.А. Шестопапов. – Аспект, 2001. – 111 с.
7. Компьютеры: справ. рук. в 3-х томах.; под. ред. Г. Хелмса. – М.: Мир, 1986.
8. Руденко В.Д. Практичний курс інформатики / В.Д.Руденко, О.М.Марчук, М.О. Патланжоглу. – К.: Фенікс, 2000. – 304 с.
9. Камарда Билл. Использование Microsoft Word 2002. / Билл Камарда. Специальное издание: пер. с англ. – М.: Вильямс, 2002. – 832 с.
10. Рыбаков М.А. Анатомия персонального компьютера / М.А. Рыбаков. – М., Интермеханика, 1990.
11. Симонович С. Новейший самоучитель по работе в Интернет / С.Симонович, Г.Евсеев. – М. ДЕСКОМ, 2000 – 528с.
12. Таненбаум, Эндрю С. Современные операционные системы. 2-е изд. – СПб. Питер, 2007.
13. Олифер В.Г., Олифер Н.А. Сетевые операционные системы – СПб.: Питер, 2008.
14. Таненбаум, Эндрю С. Архитектура компьютера. 5-е изд. – СПб. Питер, 2010.

15. Таненбаум, Эндрю С. Операционные системы. Разработка и реализация. – СПб. Питер, 2006.
16. Гласс Г., Эйблс К. UNIX для программистов и пользователей. – СПб.: БХВ-Петербург, 2004.
17. Курячий Г. В., Маслинский К. А. Операционная система Linux. – 2005.
18. Дейтел Х. М. Операционные системы. [Т. 1] / Х. М. Дейтел, П. Дж. Дейтел, Д. Р. Чофнес ; пер. с англ. под ред. С. М. Молявко. – М., 2006. – 1023 с. : ил.
19. Гордеев А. В. Операционные системы : учебник для вузов / А. В. Гордеев. – СПб., 2007. – 415 с. : ил.
20. Фигурнов В. Э. IBM PC для пользователя. Краткий курс / Э. В. Фигурнов. – М., 2006. - 479 с. : ил.
21. Корабльов В. А. Курс лекцій: «Теорія операційних систем», «Архітектура ПК», «Операційна система Linux» - pdpu.moou.com.

Доброва Т.Г.

кандидат економічних наук, доцент
кафедри політичних наук і права

БІЗНЕС-ПЛАН ДЛЯ МАЛОГО ПІДПРИЄМСТВА

1. Призначення бізнес-плану

Будь-який бізнес, незалежно від його масштабів, націлений на отримання економічної вигоди. Однак на неї можна розраховувати, якщо створюване підприємство буде пристосовано до економічних умов конкретного ринку, певного регіону, країни. Аналіз можливостей такого проникнення в ринок і є бізнес-планування.

По своїй сутності **бізнес-план являє собою економічний проект, програму підприємницької діяльності**. Він описує підприємство (вже існуюче або майбутнє); плановану продукцію або послуги (особливо важлива перевага в порівнянні з конкурентами); ринок, на який ця продукція (послуги) націлені; ресурси й кваліфікацію персоналу; фінансові потреби й плановані фінансові результати.

Бізнес-план як *проект*, як обґрунтування нового підприємства є обов'язковим етапом його створення. Він необхідний для будь-якого бізнесу, як малого, так і великого. Бізнес-план – це дорожня карта на шляху до успіху. Це креслення, по якому ви будете створювати своє підприємство

Але бізнес-план як спеціально оформлений, узагальнюючий документ складається у двох випадках:

- якщо підприємець розраховує **залучити партнерів** (інших підприємців з їхнім досвідом і фінансами) для спільного ведення бізнесу;
- якщо підприємець розраховує для реалізації свого проекту **одержати фінансову підтримку** із зовні (у вигляді кредитів, інвестицій, устаткування в лізинг, податкових пільг і ін.).

Бізнес-план надає власникові фінансових ресурсів інформацію для відповідних переговорів і демонструє обґрунтованість підприємницького задуму.

Важливо мати на увазі, що організації, що надають фінансову підтримку, одержують безліч проектів, тому за інших рівних умов перевага буде віддана більш проробленому й більш обґрунтованому бізнес-плану.

Однак не слід думати, що підприємець, який розраховує лише на власні ресурси, може обійтися й без бізнес-плану. **Для внутрішнього використання він не менш корисний.**

Розробляючи бізнес-план для реалізації бізнес-ідеї, підприємець продумує свою ринкову стратегію, порівнює свої задуми з реальністю й різними ресурсними обмеженнями. Це дозволяє при реалізації задуму уникнути таких фундаментальних помилок, як нестача капіталу для створення й функціонування підприємства й негативний баланс руху грошових