**Загальні вимоги до виконання комп’ютерного практикуму**

## Обладнання

Для виконання практикуму вам потрібен комп'ютер, на якому встановлена операційна система із сімейства UNIX. Передбачається виконання робіт в комп’ютерній лабораторії ФТІ на системі Linux з використанням технології віртуальних машин.

Для самостійної підготовки рекомендуються вільно (безкоштовно) розповсюджувані операційні системи з сімейства Linux (наприклад, Debian, Ubuntu, або інші). Можна виконувати роботи також на системах BSD (наприклад, FreeBSD), але слід пам’ятати, що вони мають певні відмінності від Linux. Для виконання окремих завдань будуть необхідні права root. Через обмеження, що встановлені політикою безпеки, система Mac OS X може не дозволити виконання деяких завдань, тому її використання для цього практикуму не рекомендується.

Зверніть увагу, що для виконання робіт Вам не потрібна багатовіконна система Х (X Window System), як і інші графічні системи (Gnome, KDE).

Серйозних вимог до апаратного забезпечення немає. Якщо не використовувати графічних систем, не буде проблем навіть із дуже застарілим комп’ютером.

## Вимоги до виконання робіт

Методичні вказівки до кожної роботи складаються з завдання для самостійної підготовки й завдання для виконання. Завдання для самостійної підготовки може виконуватись без комп’ютера (якщо є роздруковані матеріали довідкової системи). Воно передбачає ознайомлення з теоретичним матеріалом і завданням до виконання, а також підготовку до виконання практичної частини роботи.

Деякі теоретичні відомості містяться безпосередньо в методичних вказівках. Але не слід обмежуватись цією далеко не повною інформацією. Зверніться до рекомендованої літератури. Багато додаткових відомостей можна знайти в Інтернеті (пошук варто починати з www.google.com, а перегляд результатів пошуку варто починати з Вікіпедії). Докладні довідкові дані по кожній (майже кожній) команді UNIX/Linux можна одержати, набравши на клавіатурі man

<необхідна\_команда>.

Роботи виконуються в комп’ютерній лабораторії в присутності викладача. При виконанні роботи студент повинен фіксувати все, що відбувається на екрані монітора комп’ютера.

Зверніть увагу на те, що в індивідуальних завданнях часто зустрічаються певні слова, рядки, вирази в кутових дужках. Це означає, що ви повинні підставити замість цього виразу разом із дужками значення цього виразу. Наприклад, якщо вас звуть Сергій, то замість студент\_<ваше ім’я> ви повинні написати (набрати на клавіатурі) студент\_Сергій.

У деяких випадках вам будуть запропоновані **варіанти** завдань, обирати з яких необхідно той, що відповідає номеру вашої залікової книжки (або той, що вказав викладач). Не намагайтесь удавати з себе бравого солдата Швейка і виконувати лише ті **пункти** завдання, які визначаються номером вашої залікової книжки. Виконати необхідно **усі пункти без виключення** (якщо викладач не вказав інше), а в окремих пунктах може бути явно вказано, що слід обрати один з **варіантів**.

Для захисту роботи студенти повинні надати викладачу оформлений звіт, вимоги до оформлення якого викладені далі. Також студент повинен дати правильні відповіді на додаткові запитання за матеріалами цієї лабораторної роботи. Зверніть увагу на те, що студент повинен володіти всім теоретичним і практичним матеріалом, а не тільки тією його частиною, яка безпосередньо входила до індивідуального завдання.

## Вимоги до оформлення звіту

Звіт повинен містити:

1. назву роботи, тему і мету;
2. номер варіанту і зміст індивідуального завдання;
3. дату і місце виконання роботи, коротку характеристику системи, на якій було виконано роботу (клас комп’ютера, версію ОС тощо);
4. протокол виконання роботи, де чітко позначено, що вводить користувач з клавіатури, і що в результаті з’являється на екрані (необхідно показати всі результати; якщо іде запис даних в файл, вміст файлу також слід роздрукувати);
5. (**обов’язково!!!**) висновки по роботі, які повинні містити саме висновки, які студент може зробити в результаті виконання цієї роботи, а не констатацію факту "в результаті виконання роботи я навчився…"; у висновках може бути зроблена оцінка наявності певних функцій операційної системи, якості або гнучкості їх реалізації, зручності інтерфейсу тощо;
6. на кожній сторінці (в верхньому або нижньому колонтитулі) прізвище студента і номер групи або залікової книжки.

Звіт з кожної роботи повинен мати титульний лист, оформлений згідно стандартних вимог, як у наведеному далі прикладі.

|  |
| --- |
| Міністерство освіти і науки України  Технічний фаховий коледж ЛНТУ  Системне програмування  Комп’ютерний практикум  Робота №\_\_\_  Виконав студент гр.<№групи><прізвище ініціали>  Перевірив  <прізвище ініціали викладача>  <поточний рік> |

# Загальні відомості про UNIX-подібні системи

Системи, які узагальнюють ознакою “UNIX-подібні”, є багатозадачними багатокористувацькими системами з розділенням часу. Насправді, це досить різні системи, які зберегли подібний інтерфейс користувача і стандартизований програмний інтерфейс (інтерфейс системних викликів), а також схожі принципи побудови файлових систем.

Всі ці системи є нащадками однієї системи, яку було розроблено на початку 70-х років ХХ століття, і яка отримала назву UNIX. Шляхи розвитку цієї системи спочатку розійшлися, потім було проведено велику роботу по відновленню сумісності систем. Тепер розрізняють дві гілки розвитку систем UNIX: SVR4 (System 5 Release 4) розробки підрозділів AT&T і BSD компанії Berkeley Software Distribution. Правом на торгову марку “UNIX” володіла AT&T, потім ці права (разом з патентами і правами на вихідні коди системи) були продані Novell, а далі – SCO (Santa Cruz Operation). Інші системи мають власну назву, що, як правило, дещо співзвучна UNIX (AIX, Minix, Xenix, Irix тощо). Коли бажають підкреслити, що мають на увазі різні операційні системи, що мають спільні риси, часто називають їх системи \*Nix.

Фактично третім напрямком розвитку UNIX-подібних систем з початку 1990-х років стала система Linux, яка з самого початку була спробою незалежно, без застосування будь-якого захищеного майновими правами програмного коду, відтворити основні риси системи UNIX для застосування її на персональних комп’ютерах. Користуючись підтримкою численних прихильників програмного забезпечення з відкритим кодом, Linux набула великої популярності, а разом з тим – і значного технологічного удосконалення. Слід визнати, що за останні роки Linux залишила позаду більшість класичних UNIX-систем, майже всі з яких фактично зійшли зі сцени. Гнучкість Linux, значною мірою зумовлена відкритим програмним кодом, дозволяє успішно застосовувати її не лише на персональних комп’ютерах, для яких вона була створена, а й на великих навантажених серверах, і на потужних обчислювальних кластерах, і у мережному обладнанні (маршрутизаторах), і у смартфонах, і навіть у вбудованих пристроях керування (як, наприклад, у деяких сучасних телевізорах).

Найбільш помітні для користувача відмінності \*Nix-систем:

* традиційні командні інтерпретатори – sh, ksh для

SVR4; csh, tcsh для BSD; bash для Linux і MAC OS X;

* файлові системи – S5 для SVR4, UFS для BSD, ext4 для Linux.

Також є численні відмінності в іменах і розміщенні важливих конфігураційних файлів. Найбільш відомі операційні системи, що їх можна віднести до SVR4: Solaris (розроблена Sun Microsystems, тепер належить Oracle), IRIX (Silicon Graphics), AIX (IBM), SCO UNIX. За зовнішніми ознаками до SVR4 також близька система Linux. До BSD відносяться в першу чергу системи розробки власне BSD, в тому числі відкриті проекти OpenBSD і FreeBSD. Системи BSD також покладено в основу MAC OS X.

Для того, щоб користувач міг розпочати роботу в системі, він повинен проідентифікувати себе і підтвердити ідентифікацію паролем. Для цього адміністратор повинен заздалегідь створити в системі обліковий запис (іноді кажуть – бюджет, англ. – account) цього користувача. Інформацією, що ідентифікує користувача є login або userid (login – це ім'яідентифікатор з одного слова, який використовується при взаємодії користувача з системою, userid – числовий ідентифікатор користувача, яким користується система). Інформацією, що автентифікує (підтверджує тотожність) користувача є пароль (рядок символів, на який у залежності від конфігурації накладаються деякі обмеження).

Після входу в систему користувач опиняється в графічному багато-віконному середовищі. Щоб потрапити до командної оболонки (shell), в якій безпосередньо будуть виконуватися всі лабораторні роботи, слід запустити термінальне вікно – Terminal. Оболонка може запускати на виконання файли, що містять програмний код. Також вона може виконувати послідовність команд, що міститься в текстових файлах – так званих пакетних файлах, файлах сценаріїв, або, як їх ще називають, скріптах (від англ. – script). До складу операційної системи, крім ядра, входить велика кількість окремих програмних файлів – системних утиліт. Переважна більшість команд системи UNIX є так званими зовнішніми командами, тобто фактично вони означають запуск тієї чи іншої утиліти. Утиліти використовують програмний інтерфейс системи UNIX (інтерфейс системних викликів) і звертаються до внутрішніх структур ядра системи. Вони забезпечують гнучкі можливості керування файловими системами, процесами, контролю стану системи.

Докладну довідку щодо кожної команди UNIX ви можете отримати використовуючи довідкову систему man, давши команду:

man <команда>

Довідковий матеріал розбито на розділи. В описах (як у довідковій системі, так і в літературі) прийнято супроводжувати посилання на певну команду, системний виклик, функцію, номером відповідного розділу довідкової системи. Наприклад, man(1), passwd(1), passwd(4). Нумерація розділів трохи різна в SVR4 і BSD:

|  |  |  |
| --- | --- | --- |
| **Розділ** | **BS D** | **SVR4** |
| Команди загального призначення (довідник користувача) | 1 | 1 |
| Системні виклики (довідник програміста) | 2 | 2 |
| Бібліотечні функції (довідник програміста) | 3 | 3 |
| Інтерфейси ядра (довідник програміста) | 4 | 7 |
| Формати конфігураційних і системних файлів | 5 | 4 |
| Різнобічна інформація | 7 | 5 |
| Системні утиліти (довідник системного адміністратора) | 8 | 1М |

Докладнішу інформацію про історію системи UNIX, інших UNIX-подібних систем і, зокрема, Linux, можна отримати з книг, наприклад, [1].

# Програмне забезпечення з відкритим кодом

Програмним забезпечення (ПЗ) з відкритим кодом називають таке ПЗ, яке:

* вільно розповсюджується;
* доступно у вигляді вихідного коду (тексту програм);  дозволено модифікувати.

Крім цього, існує ПЗ, що використовує відкриті стандарти (тобто стандарти, які вільно розповсюджуються і опис яких доступний). Таке ПЗ може не мати відкритого коду, але при цьому розробникам іншого ПЗ легко забезпечити взаємодію з ним, тому що стандарти і протоколи обміну даними оприлюднені і доступні усім.

Також розрізняють безкоштовне ПЗ, яке також називають вільним (англ. – free) за принципом розповсюдження – вільно, без оплати.

ПЗ з відкритим кодом користується заслуженою популярністю в усьому світі, оскільки воно:

* гарантує незалежність користувача від виробників (вихідний код залишається у користувача навіть якщо виробник перестає підтримувати свій продукт);
* дає можливість кожному користувачу особисто переконатись у відсутності програмних закладок (стороннього щодо проголошеної функціональності або шкідливого коду);
* має більшу гнучкість: користувач може модифікувати вихідний код для адаптації ПЗ до своїх потреб як самостійно, так і доручивши цю роботу незалежним розробникам;
* у деяких випадках є дешевшим в експлуатації: початкова вартість може дорівнювати нулю, вартість підтримки зазвичай нижче, ніж вартість підтримки закритого ПЗ, широке відкрите для спілкування співтовариство користувачів дозволяє витрачати менше грошей на навчання і вирішення проблем.

У світі поширені найрізноманітніші програмні пакети, і одна з проблем тих, хто ними користується – сумісність форматів, в яких ці програмні пакети зберігають дані. Сучасне ПЗ як з відкритим, так і з закритим кодом зазвичай розробляють так, щоби воно було сумісним за форматами даних як з відкритими програмними пакетами, так і з розробками компаній, що продають свої продукти з закритим кодом.

Наприклад, пакет офісних програм Libre Office, функціонально дуже близький до інших популярних офісних пакетів, застосовує для зберігання даних відкритий стандарт, але допускає збереження і імпортування даних у закритих форматах інших офісних пакетів.

Існують десятки тисяч програмних пакетів з відкритим кодом; однак значний вплив на розвиток і поширення сучасних технологій мають лише деякі з них. Серед них слід відзначити операційні системи з відкритим кодом. Більшість таких систем походять від операційної системи UNIX. Серед них у першу чергу слід назвати Linux, яка значною мірою визначає шлях розвитку усього відкритого ПЗ, також відкритими є кілька систем з гілки BSD (FreeBSD, OpenBSD), а протягом певного часу до відкритих належала і Solaris від Sun Microsystems (теперішні власники цієї системи не підтримують її відкритість, а й взагалі проголосили про згортання її підтримку).

Укладачі цього комп’ютерного практикуму закликають студентів вивчати і широко застосовувати відкриті програмні продукти – як операційні системи, так і прикладні програми.