***Тема заняття: «Робота зі стрічками в мові С»***

***Мета заняття:***

* **навчальна:** ознайомити студентів із поняттям стрічки у середовищі С. Набути практичних навичок роботи зі стрічками .
* **розвивальна:** розвинути інтерес до предмету, стимулювати логічне мислення та пізнавальну активність студентів;
* **виховна**: виховувати культуру ведення математичних записів, дисциплінованість, любов до предмету.

***У результаті вивчення теми студенти повинні:***

***знати:***

* основні поняття та означення рядків;
* основні функції для опрацювання рядків;

***вміти:***

* будувати алгоритм для розв’язування задач з використанням рядків;
* розв’язувати задачі з використанням функцій для опрацювання рядків.

***Форма заняття:*** лекція.

***Тип лекції:*** інформаційна.

***Обладнання:*** проектор та екран.

***Наочність:*** Комп’ютерна презентація з теми «Робота зі стрічками в мові С».

***Базові поняття й терміни:*** стрічка, рядок, функція, масив, алгоритм.

***Перелік літератури:***

### Литвиненко Н. А. - Технология программирования на С.

### Энтони Уильямс-Параллельное программирование на С++ в действии.

### Р. Лафоре - Объектно-ориентированное программирование в С++.

***Хід заняття***

***… Питання про те, чи вміють комп’ютери думати,***

***схоже на питання — чи вміють плавати підводні човни.***

Едсгер Дейкстра

**Ι. Організаційний момент (2 хв.).**

**ΙΙ. Актуалізація опорних знань, перевірка засвоєних знань під час минулого заняття (10 хв.).**

Перед тим, як перейти до нової теми дізнаємось наскільки добре ви засвоїли тему, яку розглядали на минулому занятті, а саме: «Масиви в мові С». Для цього проведемо невелику **гру «Перевірка сусіда»:**

**Правила гри:**

*Групу ділимо на дві команди. По черзі кожен учасник команди задає запитання учаснику з протилежної команди, обираючи кожен раз іншого. Якщо студент відповів, то команда отримує 1 бал, якщо ні – відповідає будь-хто інший з команди, в такому випадку команда отримує 0,5 бала. Якщо жоден учасник з команди не відповів, відповідає студент, який задавав питання. Запитання задає по черзі кожна команда. Виграє та команда, яка набрала більше балів.*

*(В процесі гри запитання висвітлюються на слайдах.)*

**ΙΙΙ. Підведення підсумків перевірки засвоєних знань (1 хв.).**

З результатів опитування робимо висновок, як студенти засвоїли пройдений матеріал і чи готові до опанування нового.

**ΙV. Повідомлення теми та мети уроку, мотивація навчальної діяльності студентів (2 хв.).**

**Тема заняття:** «Робота зі стрічками в мові С».

**Мета заняття:** Ознайомитись із поняттям стрічки у середовищі С. Набути практичних навичок роботи зі стрічками.

**V. Виклад нового теоретичного матеріалу (40 хв.).**

В мові програмування **С** рядок символів розглядається як масив елементів типу char, який закінчується **символом ‘\0’** (нуль символ), що є ознакою кінця рядка. Такі рядки називають **ASCII-рядками**. Сталі типу рядок записують у лапках, наприклад **“Технічний коледж”, “студенти”, “ ”** – рядок, що містить один символ пропуск.

**Зауваження!** Більшості компіляторів мови С автоматично додає нуль-символ, тому зазначати його не обов’язково.

**Масиви рядків оголошуються так:**

**сhar**<назва рядка>[довжина рядка]

**Приклад**:

**const char** text1[] = “Ми вивчаємо програмування”**;**

**сhar** text2[] = “Коледж”;

**char** m1[7],m2[25];

**Символьний масив text2 можна оголосити ще так:**

**сhar** text2[7] = “Коледж”;

**або так:**

**сhar** text2[] = {‘К’, ‘о’, ‘л’, ‘е’, ‘д’, ‘ж’, ‘\0’};

**Рядки можна опрацьовувати посимвольно за допомогою вказівників або назви масиву, наприклад, так:**

**for** (int n=0; n<7; n++) (m1+n)= \*(text2+n);

**printf(“%s**”,m1);

Змінній m1 надається значення “Коледжу” і ця фраза виводиться на екран.

У бібліотеці **conio.h** визначені стандартні функції введення-виведення рядків. (**getc(), getchar()** – зчитують по одному символу рядка, введеного з клавіатури, **putc() та putchar()** – виводять окремі символи рядка).

У бібліотеці **stdio.h** описані функції для введення **gets()** та виведення **puts()** усього рядка.

**Функції для опрацювання рядків:**

**strlen(<рядок>)** – визначає фактичну кількість символів у рядку, застусовується у виразах;

**strcat(C, r2)** – команда з’єднання рядків C, r2 в один рядок, результат присвоює змінній С;

**strncat(C, r2, n)** – до змінної С додає перших n символів рядка r2 в рядок С, команда;

**strcpy(C, r2)** – копіює символи з рядка r2 в рядок С, команда;

**strcpy(C, r2, n)** – копіює перших n символів рядка r2 в рядок С, команда;

**strchr(C, <символ>)** – визначає перше входження деякого символу у рядок С так: повертає рядок, який починається від першого входження заданого символу до кінця рядка r2, застосовується у виразах;

**strrchr(C, <символ>)** – визначає останнє входження заданого символу у рядок, застосовується у виразах;

**strspn(C, r2)** – визначає номер першого символу, який входить у рядок С, але не входить у рядок r2, застосовується у виразах;

**strstr(C, r2)** – визначає у рядку С підрядок, що починається з першого входження рядка r2 у рядок С, застосовується у виразах.

**strtok(C, r2)** – визначає частину рядка С, яка закінчується перед першим однаковим символом рядків С та r2;

**strnset(C, <символ>, n)** – вставляє n разів заданий символ перед рядком С, застосовується у виразах;

**strupr(C)** – перетворює усі малі літери рядка у великі;

**strlwr(C)** – перетворює усі великі літери рядка на малі;

**strrev(C)** – записує рядок у зворотньому порядку.

Розглянемо результат застосування функцій до таких змінних:

**char** A[]= “Технічний коледж”,

 B[30]= “ТДТУ”,

 C[30]=””;

**Char \*p; int** n**;**

|  |  |
| --- | --- |
| Застосування функцій | Результат |
| n = strlen(A) | n=15 |
| strcat(B,A) | B=” ТДТУ Технічний коледж” |
| strncat(B,A,9) | C=” ТДТУ Технічний” |
| strcpy(C,A) | C=” Технічний коледж” |
| strncpy(C,A,9) | C=” Технічний” |
| p= strchr(A,K) | p=”коледж” |
| P=strrehn(A,’e’) | p=”кДж” |
| n=strson(A, “Техніка”) | n=5 |
| p=strstr(A, “кол”) | p=”коледж” |
| p=strtok(A, “КЛ”) | p=” Технічний” |
| P= strnset(A, ‘x’,10) | p=”хххххххххколедж” |
| p=strupr(“I Love You”) | p=” і Іove уou” |
| p=strlwr(“I Love You”) | p=” I LOVE YOU” |
| p=strew(“коледж”) | p=” жделок” |

Зауваження! Функції перетворення літер strlwr і strupr діють лише для латинського алфавіту. У бібліотеці stdlib.h є стандартні функції перетворення типів даних. Зокрема функція atoi(С) перетворює рядок символів C у дане цілого типу int, а функція itoa(<числове дане>, C, <система>) – дане цілого типу int у рядок С. Для перетворення даних типу double у рядок символів визначена функція gevt(<числове дане>, <кількість знаків у числі>, C), а обернену дію виконує функція strtod.

Розглянемо результати дії цих функцій для оголошених нижче змінних.
 int n;
 double f;
 char C[5],\*p;

|  |  |
| --- | --- |
| Застосування функцій | Результат |
| n=atoi(“10”) | n=10 |
| Itoa(12,C,10) | C=”12” |
| gevt(-3,14,4,C) | C=-3.14 |
| f=strtod(“-3.1415”,&p); | f=-3.1415 |

Рядки символів можна порівняти між собою. Два рядки порівнюють зліва направо посимвольно, причому ‘A’ < ‘B’, ‘B’<’C’ тощо. Більшим уважається символ, який розміщений в алфавіті дальше (він має більший номер у таблиці кодів ASCII, див. далі). Для порівняння рядків у модулі string.h надані такі функції:

 strcmp(C,r2) – порівнює рядки символів С і r2 з урахуванням регістра для латинського алфавіту;

 strcmp(C,r2) – порівнює рядки С і r2, не розрізняючи великих і малих літер латинського алфавіту.

Результатом виконання цих функцій є від’ємне число (якщо рядок її менший від рядка r2), 0 (якщо рядки однакові) або додатне число (рядок C більший рядок за рядок r2).

 Розглянемо результат виконання наступних функції

|  |  |
| --- | --- |
| Функція | Результат |
| n=strcmp(“Весна”,”весна”) | n=-32 |
| n=strcmp(“весна”,”Весна”) | n=32 |
| n=strcmp(“Весна”,”Весна”) | n=0 |
| n=strcmp(“Весна”,”весна”) | n=32 |
| n=strcmp(“Vesna”,”vesna”) | N=0 |

Розглянемо програму пошуку слова у фразі Нехай задано рядок “Скоро будуть канікули”. Визначити довжину рядка. Вивести на екран друге слово цього рядка.

#include <sdio.h>
#include <string.h>
main()
{
char C [] =” Скоро будуть канікули”;
char\*p:
printf(“%s”,C);
printf(“%d”,strlen(C));
p=strchr(C,””);
strtok(p,””);
printf(“%s”,p);}

Розглянемо ще один спосіб, в яку рядок розглядається як масив символів для роботи з рядками.

# include<sdio.h>
# include<conio.h>
# include<string.h>
main()
{
clrscr();
char C [] =” Скоро будуть канікули”;
char\*p:
int n1, n2, k, m;
m=0;
k=strlen(C);
printf(“%d”,k)
for (int i=0; i<k; i++)
if(C)[i]==”)
{
m++
if(m==1) n1=i;
if(m==2)n2=I;
}
for(i=n1+1;i<n2-1;i++)
getch();
}.

**VΙ. Систематизація та узагальнення матеріалу (12 хв.).**

Для визначення рівня засвоєння вивченого матеріалу кожен із студентів(по порядковому номері в списку жулналу) отримує індивідуальне завдання. Завершивши роботу, студент демонструє викладачеві отриманий результат. Правильна відповідь відповідно оцінюється.

**VΙΙ. Підсумкова частина заняття (10 хв.).**

1. Для узагальнення засвоєння матеріалу проведемо **бліц-опитування** з використанням **Інтерактивної гри «Так-Ні»** *(Запитання для гри висвітлюються на слайдах).*

*Гра проводиться за наступними правилами: викладач формулює твердження, а хтось із студентів погоджується – «Так», або заперечує – «Ні». (якщо відповідь «Ні», тоді потрібно вказати правильне твердження).*

1. Оголошення студентам результатів роботи (оцінювання).

**VΙΙΙ. Рефлексія (2 хв.).**

* + 1. Чи досягли ми мети поставленої на початку заняття?
		2. Під час заняття мені найбільше запам’яталось…
		3. Що було легким, що було складним?
		4. Який у вас настрій після заняття?

 А)  Б)  В) 

**ΙХ. Домашнє завдання (1 хв.).** Опрацювати теоретичний матеріал. Розв'язати індивідуальне завдання відповідно до свого номера варіанту.