12201922 이규민

5. Homework

1) [A char constant is an ascii number. A string constant is an address where it is stored in the string area.] Explain the result for following code.

#include <stdio.h>

#include <string.h> // you need this header file for string functions

void main(){

char x, y;

x='a'; y=97;

printf("%d %c %d %c\n", x, x, y, y);

char \* x1 = "hello";

printf("%s %p %s %p\n", x1, x1, "hello", "hello"); // use %p for address

}

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

캐릭터 상수는 아스키코드를 저장하기 때문에 따옴표를 사용하거나 아스키코드를 사용해 문자를 저장할 수 있다. ‘a’의 아스키 코드는 97이므로 위 코드의 x와 y는 같은 내용이 저장된 것을 볼 수 있다.

스트링 상수는 스트링이 저장되어있는 주소를 저장한다. X1의 경우 쌍따옴표를 사용하여 스트링의 주소를 저장했다. 위 결과 사진을 보면 x1의 내용과 주소는 스트링 “hello”의 내용과 주소와 동일한 것을 볼 수 있다.

Printf 에서 %d는 아스키 코드를 그대로 출력하고, %c는 문자로 전환하여 출력한다. %s는 스트링을 출력하고, %p는 스트링이 저장된 주소를 출력한다.

2) [A char constant is an ascii number] Try following code and explain the result.

char x[10]; // x is a character array with 10 rooms

x[0]=’a’; x[1]=’b’; x[2]=’c’; x[3]=’d’; x[4]=’e’;

x[5]=’f’; x[6]=’g’; x[7]=’h’; x[8]=’i’; x[9]=’j’;

int i;

for(i=0;i<10;i++){

printf(“%d %c \n“, x[i], x[i]); // print each character with its ascii number

}

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

X 배열안에 따옴표를 이용해서 문자의 아스키코드를 하나씩 저장해주었다. X를 아스키코드와 문자로 출력해보면 그 결과는 위 사진과 같다.

3) Try below. Compare the result with that of Problem 2).

char x[10]; // x is a character array with 10 rooms

int i;

for(i=0;i<10;i++){

x[i]=i+97;

}

for(i=0;i<10;i++){

printf(“%d %c \n“, x[i], x[i]); // print each character with its ascii number

}

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

2번 문제에서는 따옴표를 이용해 문자로 아스키코드를 저장했지만, 3번 문제에서는 아스키코드(숫자)로 저장을 했다. 사실상 동일하게 아스키코드로 저장했다고도 볼 수 있다.

4) Declare a character array with 128 rooms. Store 0 to 127 in this array and print the corresponding character for each ascii code in the array. Find ASCII table in the Internet and confirm the results.

char x[128];

for(i=0;i<128;i++){

x[i]=i;

}

for(i=0;i<128;i++){

printf(“%d%c%d\n”, x[i], x[i], x[i]);

}

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 스크린샷, 텍스트, 예술, 패턴이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 평행, 번호이(가) 표시된 사진

자동 생성된 설명

0번부터 127번까지의 아스키코드를 출력해보았고, 아스키코드(10진수)와 문자를 출력해보았다. 영문자, 특수문자, 공백문자 등 아스키코드표와 동일하게 출력되는 것을 볼 수 있다.

5) [strlen] Read a string and display its length.

Enter a string

hello

The length is 5

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

scanf를 통해 입력 받은 내용을 배열 x가 가리키는 공간에 저장할 수 있도록 만들었다. Strlen 함수를 이용해서 x의 길이를 출력할 수 있다.

6) [A string is a char array ending with 0] Read a string and display each character in different lines.

Enter a string

hello

h

e

l

l

o

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

scanf를 통해 입력 받은 내용을 배열 x가 가리키는 공간에 저장해줬고, for문을 스트링의 길이만큼 돌려서 한 글자씩 출력해줬다.

6-1) [A string is a char array ending with 0] Try below and explain the result. Use g++ to compile.

char x[10];

strcpy(x, "hello");

strcpy(x, "hi");

for(int i=0;i<10;i++){

printf("%d ", x[i]);

}

텍스트, 폰트, 스크린샷, 그래픽이(가) 표시된 사진

자동 생성된 설명

문자를 아스키코드로 출력해주는 코드다. 처음에는 “hello”라는 문자를 저장해줬기 때문에 배열 x에는 104 101 108 108 111 0으로 저장이 되어있었으나 “hi”라는 문자를 저장해줬기 때문에 104 105 0 108 111 0으로 저장되었다. 메시지 입력 끝은 0이기 때문에 104 105 뒤에도 0으로 바뀐 것을 알 수 있다.

7) [strlen, strcmp] Write a program that keeps reading a string, displaying its length, and checking whether it is "hello". If the input string is "hello", the program stops.

Enter a string

hi

You entered hi. length=2

No it is not hello

Enter a string

hello

You entered hello. length=5

Yes it is hello. Bye.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

입력 받은 문자가 hello 라면 멈추고, 아니라면 계속 물어보는 코드를 만들어야한다.

무한으로 도는 for 문 안에서 Scanf를 통해 입력을 받고, strlen을 통해 스트링의 길이를 출력하도록 만들었다. For문 안에서 입력 받은 스트링이 “hello”인지 확인하고, 맞다면 for 문을 나가도록 만들었다. “hello”인지 확인을 위해 strcmp 함수를 사용했다. 이후 위 사진처럼 정상적인 결과가 나오는 것을 볼 수 있다.

8) [strcpy] Read a string and copy it to three other string variables and change the first letter of them to 'a', 'b', and 'c', respectively, and display them.

Enter a string

hello

After copying and changing the first letter

aello bello cello

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

배열 x, y ,z를 만든다. Scanf를 통해 입력받은 스트링을 x의 주소에 저장한다. 그리고 strcpy로 x를 y와 z에 복사한다. 그러면 x, y, z는

9) [string constant] A string constant such as "hello" is an address. Explain the result of following code.

char \*x, \*y, \*z;

x="hello"; y="hi"; z="bye";

printf("%s %s %s\n", x, y, z);

printf("%p %p %p\n", x, y, z);

10) [string constant is an address] Try below and explain why we have an error.

char x[20];

strcpy(x, "hello"); // this is ok

x="hello"; // this is an error. "hello" is an address and we can't store address in

// x which is not a pointer variable

11) [You need memory space for strcpy] Try below and explain why we have an error. How can you fix it?

char \*y;

y="hello1"; // this is ok

strcpy(y, "hello2"); // error because y has no space for "hello2"

12) [You need memory space for scanf] Try below and explain why you have an error. Fix it.

char \*y;

printf("enter a string\n");

scanf("%s", y); // error becuase y has no space for a string

printf("you entered %s\n", y);

13) [char pointer array] Define a character pointer array and store/display strings as below.

char \* x[10];

x[0]="hi"; x[1]="bye"; x[2]="hello"; // store addresses of string constants

x[3]=new char[50];

strcpy(x[3], "there"); // copy a string

x[4]=new char[50];

printf("Enter a string\n");

scanf("%s", x[4]); // read a strign from the user

for(int i=0;i<5;i++)

printf("%s \n", x[i]);

14) [char pointer array, strcmp, new] Write a program that keeps reading strings and store them in a character pointer array. It stops when the user enters "end" and displays all strings entered so far. Use "new" to allocate memory and use g++ to compile.

Enter a string

hi

Enter a string

aaa

Enter a string

bbb

Enter a string

end

Strings entered so far are

hi aaa bbb

15) [gets, fgets] Read the same sentence with gets() and fgets() and explain the difference. (Ignore warning for gets. It is a security warning because gets can cause security problem.)

char x[100];

printf("enter a sentence\n");

gets(x);

int slen=strlen(x);

printf("sentence length after gets:%d\n", slen);

for(i=0;i<slen;i++){

printf("%x ", x[i]);

}

printf("\nenter the same sentence\n");

fgets(x, 99, stdin); // read max 99 char’s.

slen=strlen(x);

printf("sentence length after fgets:%d\n", slen);

for(i=0;i<slen;i++){

printf("%x ", x[i]);

}

16) [strtok] Use strtok to extract words from a sentence and store them in an array. Display the number of words as below. Note that you need to copy the sentence to another string variable before doing strtok because strtok will destroy the original sentence.

algorithm:

read a line

tokenize

display tokens

Enter a sentence

aa bcd e e ff aa bcd bcd hijk lmn al bcd

You entered aa bcd e e ff aa bcd bcd hijk lmn al bcd

There were 12 words:

aa

bcd

e

e

ff

aa

bcd

bcd

hijk

lmn

al

bcd

The original sentence was: aa bcd e e ff aa bcd bcd hijk lmn al bcd

17) [char pointer array, new, strcmp] Write a program that keeps reading a name and stores it in a character pointer array until the user enters bye. The program should display all names after it sees "bye".

Enter a name

kim han kook

Enter a name

park dong il

Enter a name

hong gil dong

bye

There were 3 names.

The names were

kim han kook

park dong il

hong gil dong

18) [There is a hidden 0 at the end of a string] Try below and explain why it behaves strange. How can you fix it?

int x3;

char x2[12];

char x1[12];

x1[0]=33;

x3=44;

strcpy(x2,"abcdefghijkl");

printf("%p %p %p %d %d %s", x1, x2, &x3, x1[0], x3, x2);

19) [You need memory space for strcpy] What is wrong with the following program? How can you fix it?

int main(){

char \* strarr[10]={NULL};

strarr[0]=”hello”;

strcpy(strarr[1],”bye”);

printf("%s %s\n", strarr[0], strarr[1]);

}

20) [char pointer array, strtok, strcmp] Write a program that reads a long sentence and

displays the frequency of each word. It also prints the word that has the maximum

frequency. **Your main function should look like below. Implement each function.**

int main(){

// step 1. read a sentence into buf

char buf[100];

get\_sentence(buf);

// step 2. extract words and store in tokens array. return num of tokens

int ntok;

char \*tokens[50];

ntok = tokenize(buf, tokens);

// step 3. show tokens

display\_tokens(tokens, ntok);

// step 4. compute unique tokens into unique\_tokens array

char \*unique\_tokens[50];

int nuniqtok;

nuniqtok=compute\_unique\_tokens(tokens, ntok, unique\_tokens);

printf(“unique tokens are “);

display\_tokens(unique\_tokens, nuniqtok);

// step 5. compute freq of each unique token into freq array

int freq[50];

compute\_freq(tokens, ntok, unique\_tokens, nuniqtok, freq);

// step 6. show frequencies of each token

show\_freq(freq, unique\_tokens, nuniqtok);

// step 7. show max freq word

show\_max\_freq\_word(unique\_tokens, nuniqtok, freq);

}

Enter a sentence

aa bcd e e ff aa bcd bcd hijk lmn al bcd

You entered aa bcd e e ff aa bcd bcd hijk lmn al bcd

There were 12 words: aa bcd e e ff aa bcd bcd hijk lmn al bcd

Frequncies: aa 2 bcd 4 e 2 ff 1 hijk 1 lmm 1 al 1

The word with the max freq: bcd