**Roll No…………….. Total No. of Pages:……**

**FUNDAMENTALS OF C PROGRAMMING**

**Time allowed: 90 Minutes Max. Marks: 40**

**General Instructions:**

* **Follow the instructions given in each section.**
* **Make sure that you attempt the questions in order.**

**SECTION-A (10\*1 mark=10 marks)**

***(All questions are compulsory)***

Q1 What is the difference between a static pointer and a dynamic pointer in C language?

A) A static pointer holds a fixed address, a dynamic pointer holds a variable address \*(Correct option)

B) A dynamic pointer holds a fixed address, a static pointer holds a variable address

C) Both static pointers and dynamic pointers hold a fixed address

D) None of the above

Q2 What is the difference between a single pointer and a double pointer in C language?

A) A single pointer holds the address of a single variable, a double pointer holds the address of a single pointer \*(Correct option)

B) A double pointer holds the address of a single variable, a single pointer holds the address of a single pointer

C) Both single pointers and double pointers hold the address of a single variable

D) None of the above

Q3 What is the difference between passing a pointer as an argument and passing an array as an argument in C language?

A) A pointer holds the address of a single variable, an array holds multiple variables \*(Correct option)

B) An array holds the address of a single variable, a pointer holds multiple variables

C) Both pointers and arrays hold the address of a single variable

D) None of the above

Q4 Can you pass a pointer as a return value from a function in C language?

A) Yes \*(Correct option)

B) No

C) Only variables can be returned

D) None of the above

Q5 What is the purpose of the constant pointer in C language?

A) To hold the address of a constant variable \*(Correct option)

B) To hold the value of a constant variable

C) To compare two pointers

D) None of the above

Q6 What is the purpose of the pointer to a pointer in C language?

A) To hold the address of another pointer \*(Correct option)

B) To hold the value of another pointer

C) To compare two pointers

D) None of the above

Q7 Can you dereference a null pointer in C language?

A) Yes

B) No \*(Correct option)

C) All of the above

D) None of the above

Q8 What is the purpose of the scanf() function in C language?

A) To read input from the user \*(Correct option)

B) To write output to the screen

C) To manipulate strings

D) None of the above

Q9 What is the syntax for a do-while loop in C language?

A) while(condition){}

B) do{} while(condition) \*(Correct option)

C) for(;;){}

D) None of the above

Q10 What is the data type used to store a single character in C language?

A) int

B) float

C) char \*(Correct option)

D) double

**SECTION-B (5\*2 mark=10 marks)**

***(All questions are compulsory)***

11. What is the purpose of a nested loop in C?

a) To perform a set of operations a specific number of times

b) To perform a set of operations only once

c) To perform a set of operations for each iteration of an outer loop \*(Correct option)

d) To perform a set of operations only if a certain condition is met

12. How does the value of a global variable change inside a function in C?

a) The value of a global variable cannot be changed inside a function

b) The value of a global variable can be changed using the static keyword

c) The value of a global variable can be changed without any restrictions \*(Correct option)

d) The value of a global variable can be changed using the extern keyword

13. What is the purpose of pointers in C programming?

a) To store the memory address of a variable \*(Correct option)

b) To perform mathematical operations

c) To control the flow of the program

d) To simplify complex code by breaking it down into smaller, reusable blocks

14.What will be the output of the following code?

int arr[3][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

printf("%d", \*(\*(arr)+2));

a) 1

b) 2

c) 3 \*(Correct option)

d) 4

15.What will be the output of the following code?

char name[10] = "Hello";

char \*ptr = name;

printf("%s", ptr+2);

a) 'e'

b) 'H'

c) 'l'

d) "llo" \*(Correct option)

**SECTION-C(Coding Question) (2x5 marks=5 marks)**

Q16. Chaitanya Mom used to give him pocket money every day based on his performance throughout the day. This pocket money can be negative or positive based on his performance. If it is negative that means Chaitanya has to give money back to his mom. If the pocket money is positive that means he performs well today and his mom will give him some money. Your task is to find the maximum money Chaitanya has at any contiguous day.

**Input:**

Given an integer number N representing the number of days.

In the next line, given an array A of size N, where A[i] is the pocket money at the end of each day.

**Constraints:**

1 <= N <= 106

-1000 <= A[i] <= 1000

**Output:**

Print the maximum positive money Chaitanya has at the end of Nth day.

Sample test Cases

|  |  |  |
| --- | --- | --- |
|  | Input | Output |
| STC1 | 5  8 -10 5 6 7 | 18 |
| STC2 | 5  -2 -3 -11 -5 -10 | 0 |

**Solution 16:**

#include<stdio.h>

int max(int num1, int num2)

{

return (num1 > num2 ) ? num1 : num2;

}

int solve(int A[],int n1) {

int ans =0,x=A[0];

for(int i=1;i<n1;i++)

{

x = max(A[i],x+A[i]);

ans = max(x,ans);

}

return ans;

}

int main()

{

int n;

scanf("%d",&n);

int a[n];

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

{

scanf("%d",&a[i]);

}

printf("%d",solve(a,n));

return 0;

}

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test Case 1 | Test Case 2 | Test Case 3 |
| Input | 5  1 2 3 4 -10 | 9  -2 1 -3 4 -1 2 1 -5 4 | 3  -125 -20 -130 |
| Output | 10 | 6 | 0 |

Q17. Suppose Amit was given a job to list down all the names of students in his class. When he was done with this job, he found he made few mistakes in the entered names by putting special characters, white spaces, numbers in the names. Your job is to help Amit by writing a function which takes a string and removes all unwanted characters from the given string.

**Input:**

Mah23esh@

**Output:**

Mahesh

**Solution**

#include <stdio.h>

int main() {

char line[150];

printf("Enter a string: ");

fgets(line, sizeof(line), stdin); // take input

for (int i = 0, j; line[i] != '\0'; ++i) {

// enter the loop if the character is not an alphabet

// and not the null character

while (!(line[i] >= 'a' && line[i] <= 'z') && !(line[i] >= 'A' && line[i] <= 'Z') && !(line[i] == '\0')) {

for (j = i; line[j] != '\0'; ++j) {

// if jth element of line is not an alphabet,

// assign the value of (j+1)th element to the jth element

line[j] = line[j + 1];

}

line[j] = '\0';

}

}

printf("Output String: ");

puts(line);

return 0;

}

**Test Cases**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test Case 1 | Test Case 2 | Test Case 3 |
| Input | Mah23esh@ | 123 | \_amit\_ |
| Output | Mahesh |  | amit |

**SECTION-D (Coding Question)(1x10 mark=10 mark)**

Q18 **Write a program in C to print a half pyramid of numbers using nested for loops. The pyramid should have a base of levels**

**Sample Input**:

5 \\ level

**Sample Output**:

1

12

123

1234

12345

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test Case 1 | Test Case 2 | Test Case 3 |
| Input | 5 | 3 | 7 |
| Output | 1  12  123  1234  12345 | 1  12  123 | 1  12  123  1234  12345  123456  1234567 |

**Solution:**

#include <stdio.h>

void printHalfPyramid(int levels) {

for (int i = 1; i <= levels; i++) {

for (int j = levels - i; j >= 1; j--) {

printf(" ");

}

for (int k = 1; k <= i; k++) {

printf("%d", k);

}

printf("\n");

}

}

int main() {

int levels = 5;

printHalfPyramid(levels);

return 0;

}