SEAT NO.

NED UNIVERSITY OF ENGINEERING & TECHNOLOGY

FIRST YEAR(BACHELOR OF SCIENCE IN COMPUTER SCIENCE& INFORMATION TECHNOLOGY)
FALL SEMESTER EXAMINATIONS 2022
BATCH 2022

Time: 3 Hours

Dated:09-02-2023
Max.Marks:60

Programming Fundamentals - CT-175

- Read the questions carefully.
- Attempt all six questions.
- All questions carry equal marks.
- Write solutions with proper indentation.
- You can make appropriate assumptions if needed.

1. In each of the following code snippets, find the error(s) and **explain** the reasons. Otherwise, **show** the output. [CLO 1, Marks 10]

A	B	C

int main(){
int y[4] = {6, 7, 8, 9};
int *ptr = y + 2;
printf("%d ", ptr[1]);
}
	int main(){
	int x = 5, y = 10;
	int const *p = &x;
	*p = 15;
	printf("%d", x); }
		int main( ) {
		float *b, *c , a = 13.5 ;
		b = &a ; c = b ; //Assume that &a=9FE3
		printf ("\n%p %p %p", &a, b, c) ;
		printf ("\n%f %f %f %f %f", a, *(&a), *&a, *b, *c); }

D	**E**	**F**

int main() {
int sum;
sum = 2 + 5 / 2 + 6 * 2;
printf("%d", sum);
return 0;
}
	#define CUBE(x) x * x * x
	void solve() {
	int ans;
	ans = 216 / CUBE(3);
	printf("%d", ans); }
	int main() {
	solve();
	return 0; }
		int main(){
		int arri[] = {1, 2 ,3};
		int *ptri = arri;
		char arrc[] = {1, 2 ,3};
		char *ptrc = arrc;
		printf("sizeof arri[] = %d ", sizeof(arri));
		printf("sizeof ptri = %d ", sizeof(ptri));
		printf("sizeof arrc[] = %d ", sizeof(arrc));
		printf("sizeof ptrc = %d ", sizeof(ptrc)); }

G	**H**	**I**

int main(){
int val = 1;
do {
val++;
++val;
} while (val++ > 25);
printf("%d\n", val);
}
	int main( ){
	int CONTINUE=0;
	if (CONTINUE=5)
	printf("success");
	else
	printf("failure");
	return 0;
	}
		addmult (int a , int b) {
		int w, z ;
		w = a + b ;
		z = a * b ;
		return (w, z) ; }
		int main( ){
		int i = 3, j = 4, k, n ;
		k = addmult (i , j) ;
		n = addmult (i , j) ;
		printf("%d %d", k, n);
		return 0; }

J		

int main(){
int array[10] = {3, 0, 8, 1, 12, 8, 9, 2, 13, 10};
0 1 2 3 4 5 6 7 8 9
int x, y, z;
x = ++array[2];
y = array[2]++;
z = array[x++];
printf("%d %d %d", x, y, z); }		

2. Answer the following short questions. [CLO 1, Marks 10]
   i. **Differentiate** between the two methods that can be used for passing arguments to a function.
   ii. **Explain** the issues that can be raised if pointers are misused during dynamic memory

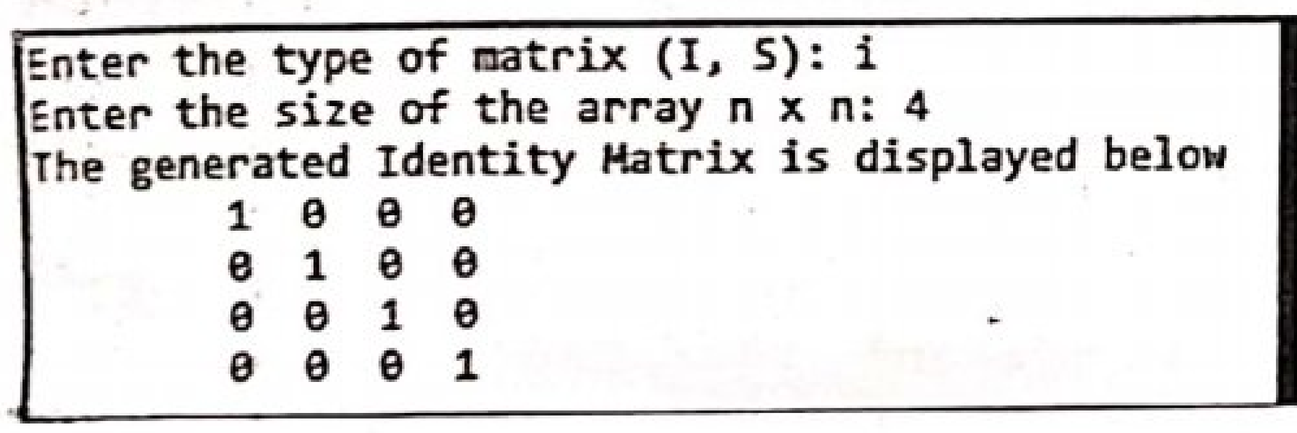
allocation.

iii. **Differentiate** between scope and lifetime of a variable and support your answer by giving suitable examples.

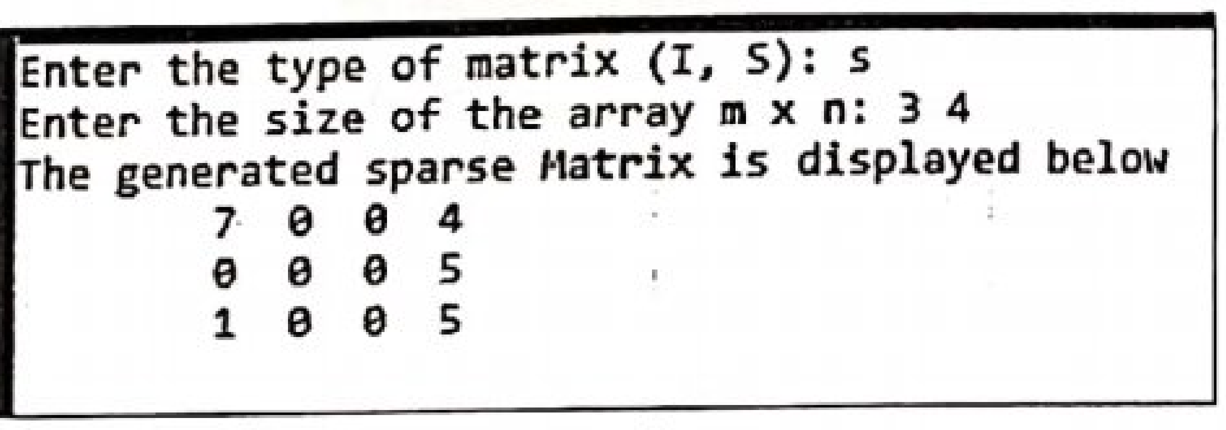
iv. **Differentiate** between compile and run-time errors. Support your answer with the help of suitable examples.

v. **Explain** the use of filing. Also, **explain** the functionality and working principle of functions fprintf() and fscanf().

3. **Outline** the steps required to solve the following problem by writing pseudo-code. Also, **illustrate** it with the help of a flow chart. You are asked to write an algorithm for creating a matrix calculator with the following features: 1) Identity matrix 2) Sparse matrix for any order of matrix. The identity matrix of size n is the ***n x n square matrix*** with ones on the main diagonal and zeros elsewhere. A sparse matrix of size ***m x n is a rectangular matrix*** in which around 67% of the elements are zero. Develop an algorithm that prompts a user to enter the type (I, S) of the matrix. Based on the input entered by the user, it asks the user to enter the size of the matrix. Declare an Array of the required size in main and then fill the array with the data so that it becomes the matrix of the type entered by the user. [CLO 1, Marks 10]



```
Enter the type of matrix (I, S): i
Enter the size of the array n x n: 4
The generated Identity Matrix is displayed below
        1  0  0  0
        0  1  0  0
        0  0  1  0
        0  0  0  1
```



```
Enter the type of matrix (I, S): s
Enter the size of the array m x n: 3 4
The generated sparse Matrix is displayed below
        7  0  0  4
        0  0  0  5
        1  0  0  5
```

4. ***Build*** *a solution in* C language for the following problem by **applying** the required programming constructs. Ask a user to input two positive numbers (integer or float) and a mathematical operator (+, -, /, *) as string, perform the required mathematical operation and return a string representation of it. [CLO 2, Marks 10]

Sample: Input: "100 * 15" Output: The product of given two numbers is 1500.
Input: "100 + 10" Output: The sum of the given two numbers is 110.

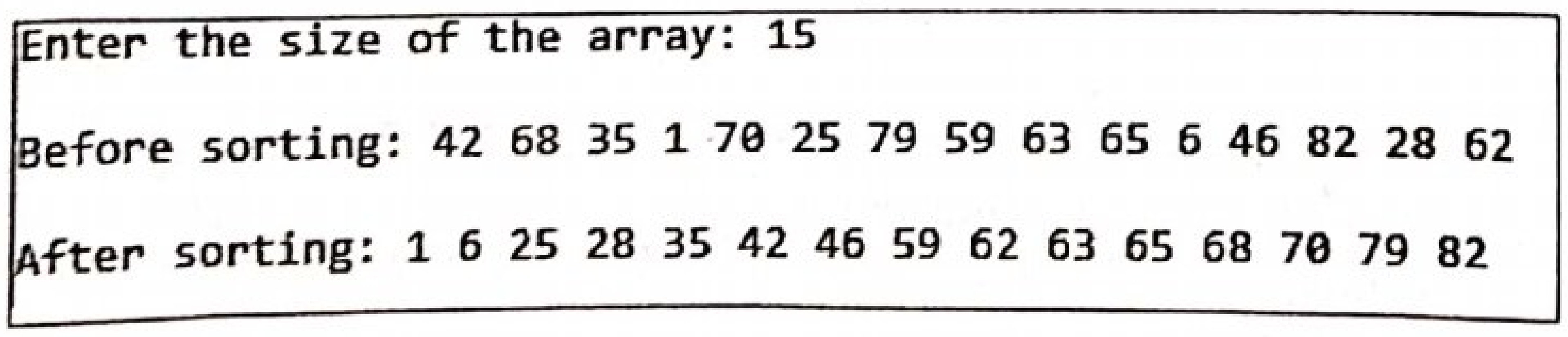
5. **Build** a solution in C language for the following problem by **applying** the required programming constructs. [CLO 2, Marks 10]

i. **Develop** a main program that prompts the user to input size of an integer array and then allocates memory on run-time on heap. Calls function ***fillArray*** for populating the array with the random data. After that it calls ***sortArray*** function for sorting the elements stored in the array. Finally, it calls the ***printArray*** function for printing the elements of the sorted Array.

ii. Develop a function: ***fillArray(int Arr[ ], int size)*** for filling array with random data between 1 to 100.

iii. Develop a function: ***sortArray(int *ptr, int size)*** which performs sorting of the array.

iv. Develop a function: ***printArray(int Arr[ ], int size)*** for printing elements of the array.



```
Enter the size of the array: 15

Before sorting: 42 68 35 1 70 25 79 59 63 65 6 46 82 28 62

After sorting: 1 6 25 28 35 42 46 59 62 63 65 68 70 79 82
```

6. **Build** a solution in C language for the following problem by **applying** the required programming constructs. ***Numeric addresses*** for computers on the Internet are composed of four parts, separated by periods, of the form ***ww.xx.yy.zz*** where ww, xx, yy, and zz are positive integers. Locally, computers are usually known by a nickname as well. You are asked to **develop** a program to process a list of Internet addresses, identifying all pairs of computers from the same locality. Construct a ***structure*** type called ***address_t*** with components for the four integers of an Internet address and a fifth component in which to store an associated nickname of ten characters. [CLO 2, Marks 10]

Sample Data

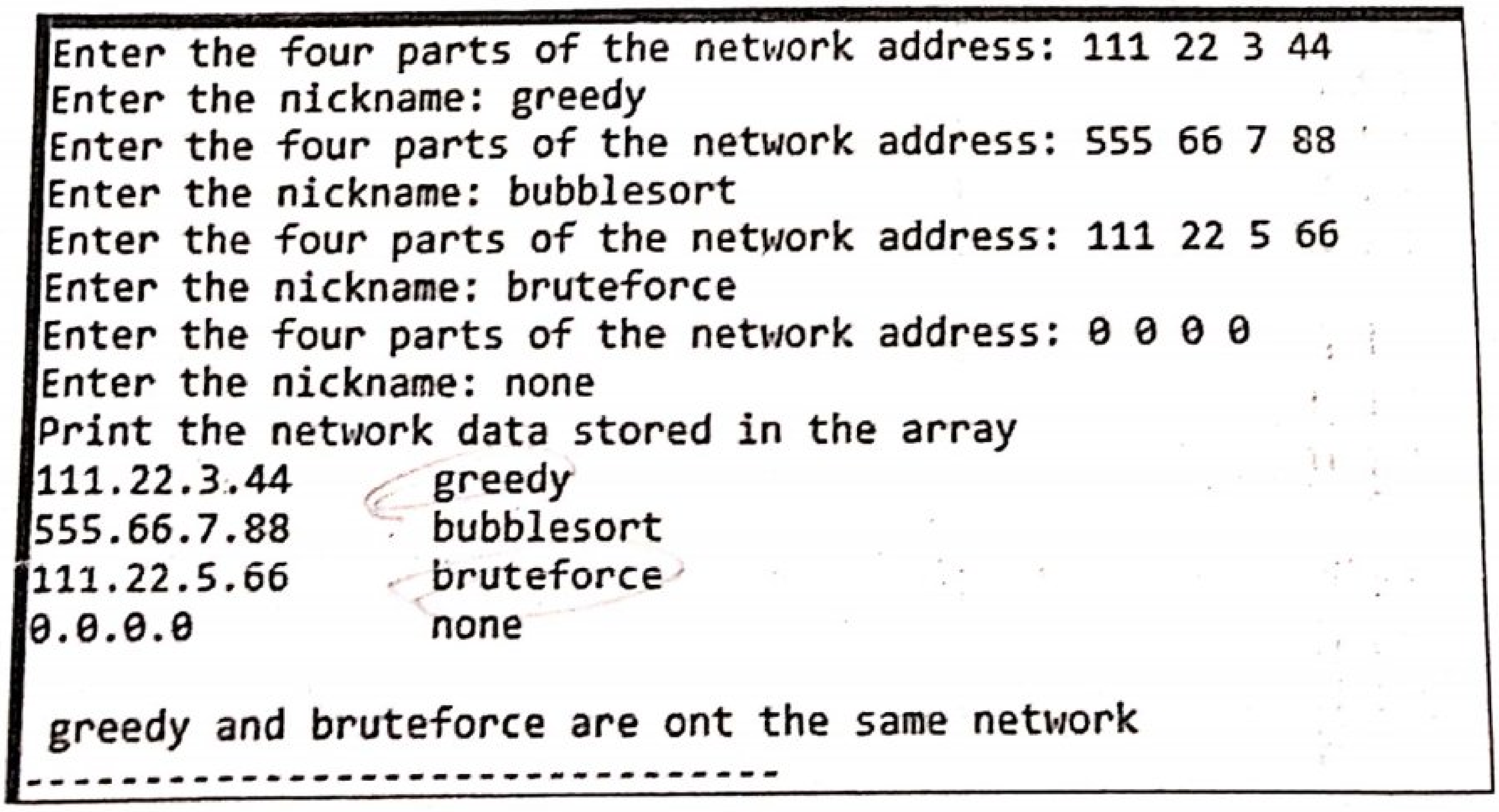
111.22.3.44	"greedy"
555.66.7.88	"bubblesort"
111.22.5.66	"bruteforce"
0.0.0.0	"none"

i. In the main function declare an array of type ***address_t.*** Your program should read a list of up to ***100*** addresses and nicknames. However, it should terminate if the address entered by a user is all ***zeros*** and the nickname is ***none.*** Call function ***getData*** for storing data in the array. After that call ***printData*** function for displaying the data stored in the array. In the end, call ***sameNetwork*** function for printing the nicknames of the networks from the same locality.

ii. Develop a function having prototype: ***address_t getData()*** for collecting data from user.

iii. Develop a function having prototype: ***void printData(address_t Arr[ ], int size)*** for printing all the network addresses and nicknames stored in the array of type ***address_t*** structure.

iv. Develop a function having prototype: ***void sameNetwork(address_t Arr[], int size)*** which should display a list of messages identifying each pair of computers from the same locality, that is, each pair of computers with matching values in the first two components of the address. In the messages, the computers should be identified by their nicknames.



```
Enter the four parts of the network address: 111 22 3 44
Enter the nickname: greedy
Enter the four parts of the network address: 555 66 7 88
Enter the nickname: bubblesort
Enter the four parts of the network address: 111 22 5 66
Enter the nickname: bruteforce
Enter the four parts of the network address: 0 0 0 0
Enter the nickname: none
Print the network data stored in the array
111.22.3.44       greedy
555.66.7.88       bubblesort
111.22.5.66       bruteforce
0.0.0.0           none

 greedy and bruteforce are ont the same network
-------------------------------
```

Date: 27/1/23

PF Final Lab Examination

Max Marks: 15

Q1. [5 marks] Declare a 2D array of size 2x3 and fill it up by taking user input. Finally, print sum of all the elements of the array.

Q2. [10 marks] Write a program that does the following:

a. Ask the user to type the size of the array.

b. Use malloc or calloc to create an integer array of that size.

c. Use the function read to read the numbers.

d. Display the average of these numbers. Show 2 numbers after the floating point in the average.

e. Free the allocated memory.

NED UNIVERSITY OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY

FS (CS, AI, DS, CY), Midterm Examinations Fall 2022

Time: 90 minutes, Date: 5-12-2022 **Programming Fundamentals (CT-175) – A** **Max Marks: 20**

Note: Attempt all questions. 5 marks on each question.

1. [CLO 1] In each of the following code snippets, find the error(s) and **explain** the reasons. Otherwise, **show** the output.

	A	B	C	D

Program	int main( ){
	char var[2]={ 'a' , 'b' };
	printf("%d", var[2]);
	}
		int main( ){
		int x;
		for(x=1; x<5; x++){
		printf("CSIT= ");
		printf("%d", x);
		if(x==4){
		continue;
		printf(" "NED= ");
		printf("%d", x); } }
		}
			int main( ){
			int x=5;
			while(x=1) {
			if(x==5){
			printf(" "NED= ");
			printf("%d", x);
			}
			}
				int main( ){
				int x , j = 1;
				for(x=1; x<5; x++){
				j = j * x;
				printf("%d", j);
				}
				}

	E	**F**	**G**	**H**

	int main ( ) {
	int x=5;
	printf("%d", x++);
	printf("%d", ++x);
	x=9;
	printf("%d", ++x);
	}
		int main( ){
		int x;
		for(x=1; x<5; x++){
		printf("*");
		if(x=5){
		break;}
		}
		}
			int main( ){
			int x=1;
			switch(x) {
			case 1:
			x=x+1; = 2
			case 2:
			x=x+1; = 3
			default:
			x=x+1; } = 4
			printf("%d", x);
			}
				int main( ){
				int @bbacus;
				for(x=1; x<5; x++){
				@bbacus = @bbacus * x;
				printf("%d", @bbacus);
				}
				}

cost > 200
more free deli delivery = 10 * i ① i = input (items) cost = ?

2. [CLO 1] With the help of a flow chart, **illustrate** step-by-step solution of the following problem. Ali typically uses the Internet to buy various items. If the total cost of the items ordered, at one time, is $200 or more, then the shipping and handling is free; otherwise, the shipping and handling is $10 per item. **Design** a flowchart that prompts Ali to enter the number of items ordered and the price of each item. The algorithm then outputs the total billing amount. Your algorithm must use a loop (repetition structure) to get the price of each item. (For simplicity, you may assume that Ali orders no more than five items at a time.)

char a, b, c, d;
if (d >= 'A' && 'Z'
d = a;
if (d >= 'a'

3. [CLO 2] Write C programs for **solving** the following problems by **applying** basic programming constructs.

a. Ask a user to input all the sides of a triangle and check whether a triangle is right-angled triangle or not. Hint: If the squares of the two shorter sides add up to the square of the hypotenuse, the triangle contains a right angle.

b. Ask a user to enter a character as input and then determine whether the given input is an alphabet or a digit or a special character by using the switch structure.

a = 'a' a >= 'a' && a <= 'z'

If (s1 > s2 && s1 > s3)
If pow(s1, 2) = pow(s2, 2)
print Δ

4. [CLO 2] Write a C program for **solving** the following problem by **applying** basic programming constructs. Ahmed and Ali are playing a game on an array A of N positive integers. Ahmed and Ali make alternating moves with Ahmed going first. In each turn, the player makes the following move:

a >= 'A' if (a >= 'A' || a <= 'Z') =
a <= 'Z'
input = a; switch (a) { case

i. If $a_0 = 0$, the element at the 0th index of the array , the player loses the game, otherwise:

ii. Player chooses an element from the array by entering its index i with 0< $i < N$. Then the player decreases the value of a_0 by 1 and swaps a_0 with a_i.

Else if (s2 >
If po
pri
Else if (s3 >
If po
prin
Else (Δ
not Δ

Test case	
N -Enter size of the array (1 < N <100,000)	3
Enter N number of integers in the array such that the sum of the elements of the array does not exceed 200,000 and $a_0 > 0$	A [3 4 1]
The game starts now.	
Ahmed turn: enter the index i of the element you have selected from the array:	1 , Status: A[4 2 1]
Ali turn: enter the index i of the element you have selected from the array:	1 , Status: A[2 3 1]
Ahmed turn: enter the index i of the element you have selected from the array:	2 , Status: A[1 3 1]
Ali turn: enter the index i of the element you have selected from the array:	2 , Status: A[1 3 0]
Ahmed turn: enter the index i of the element you have selected from the array:	2 , Status: A[0 3 0]
The game is over since $a_0 = 0$, and the winner is:	Ahmed