



PART-B

আন্তর্জাতিক ইসলামী বিশ্ববিদ্যালয় চট্টগ্রাম
الجامعة الإسلامية شيت فونخ

International Islamic University Chittagong

SL. No. 24(F)

40166

ANSWER SCRIPT



03 JUN 2024

Dept. of Electronic &
Electrical Engineering

Question No.	Marks Allocated	Marks Obtained	Remarks
1		5	
2		5 + 2.5	
3		5 + 2.5	
4		2.5 + 6	
5		4 + 4	
6			
7			
8			
9			
10			
Total		23.5 + 1.5	

Signature of Examiner

Matric/ID No. : T233003.....ID No. (In Words) : T-Two -Three -Three -Zero-Zero -Three

Semester : Spring, Autumn, Section: AAcademic Year : 2024 Mid-Term Exam, Final Exam.

Program : B.Sc in ETE Semester Enrolled : 2nd

Course Title : computer fundamental and Programming

Course Code : CSE-1221 Date of Exam : 03-06-24

Signature of Invigilator with date

No.	Serial Number of Additional Answer script	Signature of Invigilator
1		
2		
3		
4		
5		
6		

Answer to the Question NO. 3(a)

The properties of array in C are

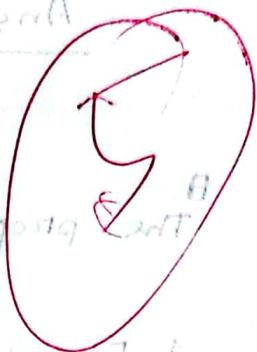
1. Fixed size
2. Zero-based indexing
3. Multidimensional Array
4. Homogeneous elements
5. Contiguous memory allocation.

This are the properties of array in C

program. Each member has its own memory address, it is fixed sized and it starts from index 0. It contains similar kind of data type. It means homogeneous elements.

Related video on multidimensional arrays
it's private and it's work with function
which will contain different functions
After compilation it's not run until we implement

```
#include <stdio.h>
int main(){
    int arr[6] = {2, 3, 3, 0, 0, 3};
    int cumulative_sum = 0;
    for(int i=0; i<6; i++){
        cumulative_sum = cumulative_sum + arr[i];
    }
    printf("The cumulative sum = %d", cumulative_sum);
    return 0;
}
```



Q. Write a program to calculate the sum of all elements in an array.

Program and output are not matching

Ans: The following is the code for the question:

Answer to the Question No. 4(a)

The following is the output of the program:

Q) The process of declaring and initializing

structure variables are given below-

Ans: At first, we have to use 'struct' to define structure before the main function. Then we have to initialized with

int, float and ~~struct char~~. If the information is integer value then we have to write int before it, if the information is ~~float~~^{decimal} value then we have to write float before it and if the information is a character then we have to write char before it. Here is an example of it,

```
struct employee{  
    int ID;  
    float salary;  
    char name;  
};
```

{Ques 5 Stri
Ques 5 Stri}

ID is a integer type value that salary is a decimal type and char name is a character type value that's why we have to use int, float, char before it respectively.

(Ans 5. [7] 8. [7])

Answer to the Question No. 4(b)

```
#include < stdio.h>
§ struct members{
    char name [20];
    char address [50];
    int age;
    char occupation [10];
    float salary;
} m[3];
int main(){
    struct members m[3];
    for (int i=0; i<3; i++){
        scanf ("%s", m[i]. name);
        scanf ("%s", m[i]. address);
        scanf ("%d", &m[i]. age);
        scanf ("%s", m[i]. occupation);
        scanf ("%f", &m[i]. salary);
    }
}
```

```

for( int i=0 ; i<3 ; i++ ) { char *b[3] > & b[0]; submit
    printf("members[%d] = %s\n", i, m[i].name);
    printf("members[%d] = %d\n", i, m[i].address);
    printf("members[%d] = %d\n", i, m[i].age);
    printf("members[%d] = %s\n", i, m[i].occupation);
    printf("members[%d] = %f\n", i, m[i].salary);
}
return 0;

```

(C++) ~~int b[3] = {1,2,3};~~ ~~b[0] = 1+1;~~

Point 1: ~~char *b[3] > & b[0];~~ submit
 Point 2: ~~int b[3] = {1,2,3};~~ ~~b[0] = 1+1;~~

Point 3: ~~int b[3] = {1,2,3};~~ ~~b[0] = 1+1;~~

Answer to the Question No. 5 (a)

Pointer :- A pointer is a variable that ~~stores~~ ~~is~~ ~~pointing~~ ~~to~~ ~~another~~ ~~variable~~ ~~in~~ ~~memory~~ ~~area~~ ~~of~~ ~~computer~~ ~~system~~
Answer :- ~~char *b[3] > & b[0];~~ submit
 stores a memory address of another variable. ~~Let's take an example of pointer~~

Hence is an example of pointer -

$$B = C^*$$

$$B = D$$

$$D = E$$

```
#include <stdio.h> // (file extension .c and .h) note
```

```
int main(){ // address = [base] + [offset] of variable
    int *y; // address = [base] + [offset] of variable
    int x; // address = [base] + [offset] of variable
    y = &x;
    *y = 5; // address = [base] + [offset]
    printf("x = %d\n", x);
    printf("*y = %d\n", *y);
```

It will print $x = 5$ and $*y = 5$. At first

we declare a pointer and variable. Then
store the address of that variable in that

pointer. $*y$ means it will print memory
address's value which is 5.

address of both statement is same, so both will print 5.

so both the memory location is same

Output

$x = 3$

$*y = 3$

$x = 5$

$*y = 5$

Answer to the Question No. 5(b)

(a) C program to upload file to memory

```
#include <stdio.h>
#include <stdlib.h>

int main(){
    int num;
    FILE *a;
    a=fopen("ete.txt","w");
    if(a==NULL){
        printf("Error");
        exit(1);
    }
    fprintf(a,"233003");
    fprintf(a,"myname.txt");
    printf("The content of the file has been
uploaded to the computer screen");
    fclose(a);
    return 0;
}
```

(4)

(d) 4 bit adder add 4 numbers

Answer to the Question No. 3(b)

Output

0 1 2 3 4

4 bit adder

Candidate's solution

for answer

given by

Output

✓ 1 2 3

✓ 4 5 6

✓ 7 8 9

3 0 5

and add 4 numbers
(Candidate's output has been checked)

7 7 7



আন্তর্জাতিক ইসলামী বিশ্ববিদ্যালয় চট্টগ্রাম
الجامعة الإسلامية - سنت فونخ
International Islamic University Chittagong

SL. No. 24(F)

40135

ANSWER SCRIPT



03 JUN 2024

Dept. of Electronic &



Question No.	Marks Allocated	Marks Obtained	Remarks
1		—	
2		—	
3		4 + 3.5	
4		0 + 4	
5		3 + 2.5	
6			
7			
8			
9			
10			
Total		17	

Signature of Examiner

Matric/ID No. : T23300 ID No. (In Words) : T-Two-Three-Three-Zero-Zero-Nine

Semester : Spring. Autumn. Section: A

Academic Year : 20.24 Mid-Term Exam. Final Exam.

Program : Bsc in ETE Semester Enrolled : 2nd

Course Title : computer Fundamentals and C-programming

Course Code : CSF-1221 Date of Exam : 3/6/2024

Signature of Invigilator with date

No.	Serial Number of Additional Answer script	Signature of Invigilator
1		
2		
3		
4		
5		
6		

Answer to the Question NO: 03

@

#include <stdio.h>

int main() {

~~int arr[N];~~

for (int

int arr[N];

scanf ("%d", &N);

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

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

}

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

 if (i>0) {

 printf ("1", arr[i]);

}

?). ??

else {

 printf ("2", arr[i]);

}

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

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

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

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

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

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

return 0; // returns int to search

f

③

(b)

Un-sibbed character

Output : 01

~~1 2 3 4 5~~

0 1 2 3 4

Y

Output : 02

1 2 3

4 5 6

7 8 9

↓

((1, "b0") frame

((2, "b1") start

((3, "b2") start

f((4, "b3") start

f(0(i) fi

((1, "b4") frame

((2, "b5") frame

((3, "b6") frame

((4, "b7") frame

Answer to the Question NO: 04

(b)

```
#include<stdio.h>

struct family f[3];
char name[20];
char address[50];
int age;
char occupation[10];
int salary;
int main()
{
    for (int i=0; i<3; i++)
    {
        scanf("%s", &f[i].name);
        scanf("%s", &f[i].address);
        scanf("%d", &f[i].age);
        scanf("%ds", &f[i].occupation);
    }
}
```

2222

```

scanf ("%d", &family [i]. salary);
for (int i = 0; i < 3; i++) {
    printf ("%s", family [i]. name);
    printf ("%s", family [i]. address);
    printf ("%d", family [i]. age);
    printf ("%s", family [i]. occupation);
    printf ("%d", family [i]. salary);
}
return 0;
}

```

Answer to the Question NO: 05

Ques: $(x + y) * z = 16$ If $x = 3$, $y = 5$, $z = ?$

~~$x = 3$~~ $i((B^*, *x/b_0) = (y_0) \text{ If } b_0 = 1)$
 ~~$*y = 3$~~ $i((B^*, *y/b_0) = (y_0) \text{ If } b_0 = 1)$
 ~~$x = 5$~~ $i((B^*, *x/b_0) = (y_0) \text{ If } b_0 = 1)$
 ~~$*y = 5$~~ $i((B^*, *y/b_0) = (y_0) \text{ If } b_0 = 1)$

pointer:

Pointer is a variable that indicates the address of a argument.

example:

int main() {

int *y;

int n;

$y = \&n;$

$*y = 3;$

printf ("x=%d\n", n);

3

printf ("y=%d\n", *y); // after backward

*y += 2;

printf ("n=%d\n", n); // through

printf ("y=%d\n", *y); $y = 30$

return 0; $x = 60$

}

B

$x = 60$

$y = 60$

$x = 60$

$y = 60$

looking

standard output function of string

branches to memory

legends

FC (function)

IG (instruction)

RE (register)

WB (write back)

DE (data exchange)

(C, N/B, DEX) Transfer

branch, jump, call, return

push, pop, mov, add, sub, mul, div, etc.

(b)

```
#include<stdio.h>
#include<stdlib.h>

FILE *file;
int main()
{
    FILE *file;
    file = fopen("file.txt", "w");
    if (file == NULL)
        fprintf("Error");
    if (fopen("file.txt", "r"))
        fscanf(
```

2.5



আন্তর্জাতিক ইসলামী বিশ্ববিদ্যালয় চট্টগ্রাম
الجامعة الإسلامية بنى فونغ
International Islamic University Chittagong



SL. No. 24(F)

40167

ANSWER SCRIPT

03 JUN 2024

Dept. of Electronic &



Question No.	Marks Allocated	Marks Obtained	Remarks
1		✓	
2		✓	
3		0+0	
4		1.5+0	No student is in favor or below in the examination
5		1+0	Examination
6			
7			
8			
9			
10			
Total		2.5	

Signature of Examiner

Matric/ID No. : T221021..... ID No. (In Words) : T-two-two-one-zero-two-one.....

Semester : Spring. Autumn. Section: A.....Academic Year : 20...24..... Mid-Term Exam. Final Exam.

Program : Bsc in ETE Semester Enrolled : 5th

Course Title : Computer Fundamentals with programming.....

Course Code : CSE - 1221 Date of Exam : 03/06/24

Signature of Invigilator with date

No.	Serial Number of Additional Answer script	Signature of Invigilator
1		
2		
3		
4		
5		
6		

Ans to the Ques No 5

①

(b)

#include <stdio.h>

int main () {

FILE *Filepointer = fopen ("elec.txt", "w");

If (Filepointer == Null) {

printf ("error to open a file")

else {

return 1

}

(*) function printf @ the 11 line contains no "f" forming

("reading file", b); here

(**) function printf @ the 11 line contains no "m" forming

(i+i, reading file); here

;("error") forming

(reading file, d); here

;("error") forming

(reading file, d); here

;("error") forming

(reading file, d); here

;("error")

Ans to the Ques No 4

(b)

```
#include <stdio.h>
```

```
struct family member family member;
```

```
char name[50];
```

```
char address[50];
```

```
int age;
```

```
char occupation[50];
```

```
int salary;
```

```
int main()
```

```
printf("Information about 3 family members:\n");
```

```
scanf("%d, &family member");
```

```
struct family member member(family member);
```

```
for (i = 0, i ≤ family member, i + +);
```

```
printf("name:\n");
```

```
scanf("%s, &family member");
```

```
printf("address:\n");
```

```
scanf("%s &family member");
```

```
printf("age:\n");
```

```
scanf("%d, &family member");
```

```
return 0 }
```

Ques No 4

Ans to the Ques No 3

(b)

The output of the code is 1, 3

Given program,

