Assignment 0

Arithmatic Operators

PAGE NO.
DATE :

Q1) C program to add two integers and disp result

⇒

memory digram with algo

1) we have to add two integers lets take 2 int type variable to store them.

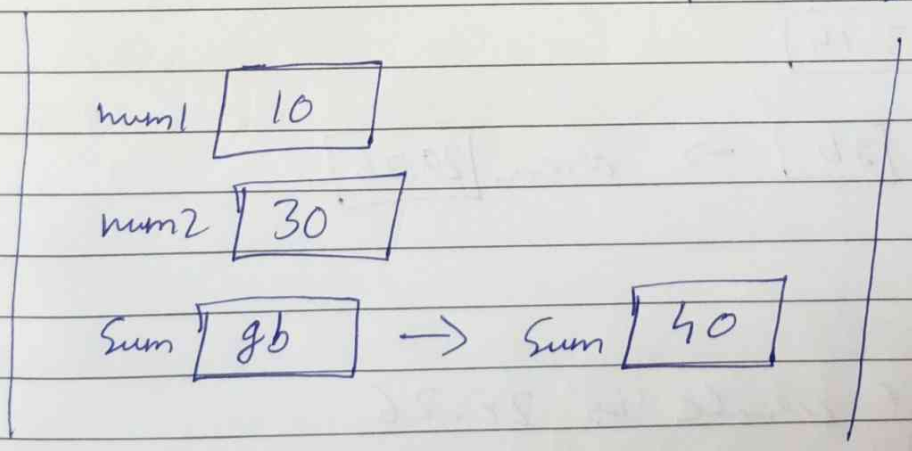
```
int num1 = 10;
int num2 = 30;
```
} initialization

2) One more variable declaration to store sum of two numbers.

```
int sum;
```
} declaration

3) `sum = num1 + num2;` } assignment

4) `printf("Addition of %d and %d is %d", num1, num2, sum);`



Output

Addition of 10 and 30 is 40

Q 2.] C program to find area of circle

→ area of circle → $A = \pi r^2$

1) According to the formula to find area we need radius so lets create one float type variable to store a ~~radius~~ radius and one variable for Pi value.

float Radius = 3.5, Pi = 3.14; } initialization

2) One more variable for storing area →

float area; } declaration

3) area = Pi * Radius * Radius; } assignment

4) printf("Area of circle is %.2f", area); }

to take values only upto 2 decimal point in output

Memory digram

Radius | 3 |

Pi | 3.14 |

area | 96 | → area | 28.26 |

Output

Area of circle is 28.26

PAGE NO.
DATE :

Q3] C program to convert temp from C to F

⇒ formula → F = (C * 9/5) + 32

1) According to the formula to convert temp from C to F we first need to create a float type variable to store temp in C.

float celsius = 3; } initiallization

2) one more floating type variable to store operational value.

float fahrenheit = ; } declaration

3) fahrenheit = (celsius * 9/5) + 32;

4) printf ("Temprature in Fahrenheit = %.1.f",
fahrenheit);

memory digram

celsius | 3 |

fahrenheit | gb | → | 37.4 |

Output

Temprature in Fahrenheit = 37.4

PAGE NO.
DATE :

Q4] Write a C program to swap two no. using a temporary third variable.

⇒ (1) First we need to create a int type variable for num1 and num2.

```
int num1 = 2
int num2 = 1
```
} initialization

(2) Now lets create ~~one~~ one temp variable of int type.

```
int temp;
```
} declaration

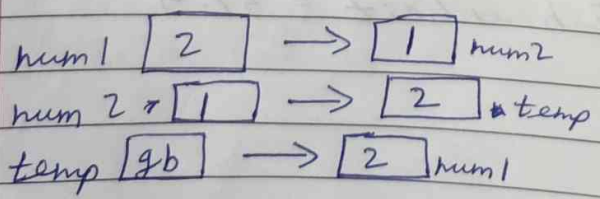
(3) Now we will perform swapping but before that lets show values of num1 and num2 before swapping ⇒

```
printf("before swapping num1 = %d and num2 = %d",
                                    num1, num2);
```

```
temp = num1;
num1 = num2;
num2 = temp;
```
} assignment.

(4) printf("After swapping num1 = %d and num2 = %d",
num1, num2);

Memory digram



output

before swapping num1 = 2 and num2 = 1
After swapping num1 = 1 and num2 = 2

PAGE NO.
DATE :

Q5. C program to input five numbers and find average.

=> (1) First we have to create five int type variables to store five number.

```
int num1, num2, num3, num4, num5;
```

declaration

```
float average;
```

(2) Now lets initiallize the five numbers

```
num1 = 10;
num2 = 15;
num3 = 16;
num4 = 20;
num5 = 21;
```

} assignment

(3) Now lets calculate the average and store it in a average variable of floating datatype which we have created in our 1st ~~step~~ step.

```
average = (num1 + num2 + num3 + num4 + num5) / 5;
```

(4) ~~printf ("Average = %f %.2f", average);~~

```
printf ("Average = %.2f", average);
```

Memory digram

num1 [gb] → 10
num2 [gb] → 15
num3 [gb] → 16
num4 [gb] → 20
num5 [gb] → 21
average [gb] → 16.4

output

Average = 16.4

DATE :

Q6] C program to find Square and cube of a given number.

⇒ (1) First lets create a integer type variable for storing a given number and another one for storing Square and one for storing cube of the given number.

```
int num = 2;   initiallization
int Square, cube;   declaration
```

⇒ (2) Now lets calculate Square and cube of a given number.

```
Square = num * num;
cube = num * num * num;
```

⇒ (3)

```
printf("Square of %d is %d", num, Square);
printf("\n cube of %d is %d", num, cube);
```

Memory digram

num [2]
Square [96] → [4]
cube [96] → [8]

output

Square of 2 is 4
cube of 2 is 8

PAGE NO.

DATE :

Q7] C program to convert given minutes into hours and remaining minutes.

=> (1) First lets create some variables of integer type for storing given minute value, one for storing hours value after converting and one for remaining min.

```
int min = 130;
int hours, rmin;
```

=> (2) Now we known that 1 hour = 60 min so if we divide a given min by 60 we will get hour as divident and remaining min as quotient.

1 + 1 = 2 hours

130 = 60 + 60 = 120

130 − 120 = 10 → remaining min

60 + 60 = 120, 120 + 60 = 180; 60 × 2 = 120

So

```
       2 — hours
60 ) 130
     120
     ---
     010 — min
```

another trial

```
       4 — hours
60 ) 250
     240
     ---
     010 — min
```

∴ hours = min / 60;

rmin = min % 60;

=> (3) printf("%d hours and %d minutes", hours, rmin);

memory digram

min [130]

hours [gb] → [2]

rmin [gb] → [10]

output

2 hours and 10 minutes

DATE :

Q8] C program to input length and width of a rectangle and find perimeter

⇒ formula ⇒ $P = 2(l + w)$

① First we need to create floating type variables for taking length and width a and store input the one more variable for storing perimeter after calculation.

```
float length = 4
float width = 5.6
```

② float perimeter; ~~= 2 (length + width);~~

```
perimeter = 2*(length + width);
```

③ printf("Perimeter of rectangle is %.2f",
perimeter);

Memory diagram

length [4]
width [5.6]
Perimeter [96] → 19.20

Output

Perimeter of rectangle is 19.20

PAGE NO.
DATE :

Q9] C program to input the base and height of ~~trian~~ triangle and calculate area.

=> $Area = \frac{1}{2} * (Base * height)$

① we need three floating type variables for storing input Base, height and Area to store operational value.

```
float base = 4 ;
float height = 10 ;
float Area ;
```

② `Area = 1/2 * (base * height) ;`
↳ 0.5 because $\frac{1}{2}$ is integer in C returns 0 only

③ `printf ("Area of triangle is %.2f", Area) ;`

Memory digram

base [4]
height [10]
Area [96] → 20.00

Output
Area of triangle is 20.00

DATE :

Q10 C program to input marks of five subjects, find the total marks, and calculate percentage.

=> (1) First need to create five integer type variables for storing marks and the one more variable for storing total marks and then one floating type variable for storing calculated percentage.

```
int sub1 = 40;
int sub2 = 65;
int sub3 = 70;
int sub4 = 80;
int sub5 = 50;
int total;
float percentage;
```

(2) ~~Enter~~ Calculating total

```
total = sub1 + sub2 + sub3 + sub4 + sub5;
```

(3) calculating percentage

~~percentage = sub1 + sub5 total~~

```
percentage = (total / 500) * 100
```

(4) Print total marks and percentage.

Memory digram

sub1 [40]
sub2 [65]
sub3 [70]
sub4 [80]
sub5 [50]
total [96] -> 305
percentage [96] -> 61