**Part-B**

1. You are tasked with devising an algorithm for a basic calculator for a small grocery store to help the cashier to calculate the total price of two items. The cashier will input the price of both items, and your algorithm should calculate and display the total cost.

Step 1: Start

Step 2: Declare two variables item\_1,item\_2, total\_price

Step 3: Input the price for item\_1, item\_2

Step 4: Calculate total price using

total\_price=item\_1+item\_2

Step 5: Display the total cost

Step 6: Stop

1. Debug and generate the desired output as return 0 or return 1.

#include <stdio.h>

int main() {

int n, i;

long sum = 0;

printf("Enter a positive integer: ");

scanf("%d", &n);

if (n < 0) {

printf("Please enter a positive number greater than zero.\n");

return 1;

}

for (i = 1; i < n; i++);

{

sum += i \* i;

}

printf("The sum of squares of the first %d natural numbers is: %d\n", n, sum);

return 0;

}

**Solution:**

#include <stdio.h>

int main() {

int n, i;

printf("Enter a positive integer: ");

scanf("%d", &n);

if (n > 0) {

printf("return 1");

}

else

{

printf("return 0");

}

return 0;

}

1. **Fill the code given and explain how the desired output will be generated.**

#include <stdio.h>

int main() {

int number;

printf("Enter a number: ");

scanf("%d", &number);

if (\_\_\_\_\_\_\_ > 0) { // Blank 1

printf("The number is positive.\n");

}

else if (\_\_\_\_\_\_\_ < 0) { // Blank 2

printf("The number is negative.\n");

}

else {

printf("The number is \_\_\_\_\_\_.\n"); // Blank 3

}

return 0;

}

**Solution:**

#include <stdio.h>

int main() {

int number;

printf("Enter a number: ");

scanf("%d", &number);

if (number > 0) { // Blank 1

printf("The number is positive.\n");

}

else if (number < 0) { // Blank 2

printf("The number is negative.\n");

}

else {

printf("The number is Zero.\n"); // Blank 3

}

return 0;

}

1. You are working on a banking application that requires a feature to help customers track their daily expenses. The program should allow users to input their daily expenses one at a time. When the user enters a negative value, it indicates that they have finished entering expenses for the day. Implement using a C program to calculate and display the total amount spent.

#include <stdio.h>

int main() {

int d\_exp;

int tot\_amt\_spent=0;

int flag=1;

while(flag)

{

printf("Enter Daily Expense\n");

scanf("%d",&d\_exp);

if(d\_exp>0)

{

tot\_amt\_spent=tot\_amt\_spent+d\_exp;

}

else

{

flag=0;

}

}

printf("Total Amount spent=%d",tot\_amt\_spent);

return 0;

}

**Part-C**

1. Design an algorithm and implement the same using C program for carrying out the following tasks

* Read yesterday’s and today’s temperature
* Find the difference

Using an operator print “Increase” if today’s temperature is higher and print “Decrease” otherwise.

**Algorithm:**

**Step 1: Start**

**Step 2: Declare two variables yes\_tempm tod\_temp**

**Step 3: find diff=yes\_temp-tod\_temp**

**Step 4: Display “increase” if today\_temp>diff.. otherwise display “decrease” using conditional operator**

**Step 5: Stop**

**Program:**

#include<stdio.h>

int main()

{

int yes\_temp,tod\_temp,diff=0;

printf("Enter Yesterday and Today's temperatures\n");

scanf("%d %d",&yes\_temp,&tod\_temp);

diff=yes\_temp-tod\_temp;

printf("\nDifference:%d\n",diff);

(tod\_temp>diff)?(printf("Increase")):(printf("Decrease"));

return 0;

}

1. A company calculates the salary of an employee based on the category of the employee. For daily wage employees, their salary is based on the number of hours worked and the rate per hour. On the other hand, regular employees have a more structured salary calculation, which takes into account several components such as basic pay, allowances, and additional benefits. For example, a regular employee’s salary includes a fixed basic pay, which is then increased based on factors like a predefined percentage of Dearness Allowance (DA), a specific House Rent Allowance (HRA), and a medical allowance. Each of these components contributes to the final salary calculation. the correct calculation is applied based on the employee type, with the appropriate salary elements factored in for each type of employee.

**Program:**

#include<stdio.h>

#define rph 40

int main()

{

int emp\_type;

int h,basic,hra,da;

int tot\_sal=0;

printf("Type-1 Employee-Daily Wages\n");

printf("Type-2 Employee-Regular Employee\n");

printf("Enter Employee Type\n");

scanf("%d",&emp\_type);

switch(emp\_type)

{

case 1:

printf("Enter Number of working hours\n");

scanf("%d",&h);

tot\_sal=h\*rph;

break;

case 2:

printf("Enter Basic Pay,HRA,DA\n");

scanf("%d%d%d",&basic,&hra,&da);

tot\_sal=basic+hra+da;

break;

}

printf("Total Salary of the EmpType-%d=%d",emp\_type,tot\_sal);

return 0;

}

3)You are developing a simple traffic light control system. The system controls the flow of traffic at an intersection. The lights can be:

Green: Vehicles can move.

Yellow: Vehicles should slow down.

Red: Vehicles must stop.

The traffic light's state is represented by three flags:

isGreen, isYellow, and isRed (all are either 0 or 1).

Write a C program that takes the states of the traffic lights as input and prints:

"Go" if the light is green.

"Slow Down" if the light is yellow.

"Stop" if the light is red. If more than one light is on or no light is on, print "Invalid signal".

Program:

#include<stdio.h>

#define rph 40

int main()

{ int isY,isG,isR;

printf("Enter state of the signal: Yellow-Green-Red\n");

scanf("%d %d %d",&isY,&isG,&isR);

if(isY==1&&isG==0&&isR==0)

{

printf("Slow Down");

}

else if(isY==0&&isG==1&&isR==0)

{

printf("Go");

}

else if(isY==0&&isG==0&&isR==1)

{

printf("Stop");

}

else

{

printf("Invalid Signal");

}

return 0;

}

4) Write a program that simulates an ATM withdrawal process. Repeatedly ask the user to enter the amount they want to withdraw until they enter a valid amount that does not exceed their account balance.

**Program:**

#include<stdio.h>

#define balance\_accounnt 500

int main()

{

int wd\_amt=0,tot\_wd=0;

printf("Balance=%d\n",balance\_accounnt);

do{

printf("Enter withdrawal Amount\n");

scanf("%d",&wd\_amt);

if(wd\_amt>balance\_accounnt || wd\_amt<0)

{

printf("withdrawal amount should be less than balance and shold not be less than zero");

break;

}

else

{

tot\_wd=tot\_wd+wd\_amt;

if(tot\_wd>balance\_accounnt)

{ printf("Amount is exceeding balance available in your account");

break;

}

printf("Total amount withdrawn=%d\n",tot\_wd);

}

}while(tot\_wd<balance\_accounnt);

return 0;

}