**Lab manual 06**

**Mosa. Rabeya**

**I’d:191-15-12145**

**Coin change - 01 (Dynamic):**

**Q:** Suppose you want to pay back someone with 16 taka and you have unlimited supply of 1 taka, 2 taka, 8 taka and 12 taka notes. Your target is to use

the minimum number of notes to payback the said amount of money. That means how many notes you chose, write the program.

**Code:**

#include<bits/stdc++.h>

using namespace std;

void coin\_change(int coin[],int totalCoin,int change)

{

int m[change+1],minimum;

m[0]=0;

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

{

minimum=change+1;

for(int j=0; j<totalCoin; j++)

{

if(coin[j]<=i)

{

minimum=min(minimum,m[i-coin[j]]+1);

}

}

m[i]=minimum;

}

if(m[change]==0)

cout<<"Change is not possible\n";

else

cout<<"Coin needed: "<<m[change]<<endl;

}

int main()

{

int i,totalCoin=4,change=16;

int coin[]= {1,2,8,12};

coin\_change(coin,totalCoin,change);

return 0;

}

**Coin change -02(Dynamic):**

**Q:** Suppose you are given the coins 1 taka, 5 taka, and 10 taka with Total Taka = 8 taka,

what are the total number of permutations of the coins you can arrange to obtain 8 taka.

**Code:** #include<bits/stdc++.h>

using namespace std;

void coin\_change(int coin[],int totalCoin,int change)

{

int m[change+1],minimum;

m[0]=0;

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

{

minimum=change+1;

for(int j=0; j<totalCoin; j++)

{

if(coin[j]<=i)

{

minimum=min(minimum,m[i-coin[j]]+1);

}

}

m[i]=minimum;

}

if(m[change]==0)

cout<<"Change is not possible\n";

else

cout<<"Coin needed: "<<m[change]<<endl;

}

int main()

{

int i,totalCoin=3,change=16;

int coin[3]={1,5,10};

coin\_change(coin,totalCoin,change);

return 0;

}

**Coin change 03(Greedy):**

**Q:** Suppose you went to a market, then you gave 15 tk note to the sales person, and he returned some of the currency (1,7,7,10S) that he had in abundance.

Then write the code which is find optimal solution.

**Code:** #include<bits/stdc++.h>

using namespace std;

void coin\_change(int coins[], int n, int m)

{

int cnt[n],i;

memset(cnt,0,sizeof cnt);

for(i=n-1;i>=0;i--)

{

if(coins[i]<=m)

{

cnt[i]=m/coins[i];

m=m%coins[i];

}

}

if(m!=0)

cout<<"Change is not possible"<<endl;

else

{

cout<<"Coin need:"<<endl;

for(i=n-1;i>=0;i--)

{

if(cnt[i]!=0)

cout<<coins[i]<<" coin "<<cnt[i]<<" times"<<endl;

}

}

}

int main()

{

int n=4,change=15;

int coins[]={1,7,7,10};

sort(coins, coins+n);

coin\_change(coins,n,change);

return 0;

}

**Coin change 04(Greedy):**

**Q:** Suppose, you have coin changing machine. Which has huge amount of coin. Now if any one gave some money then it is return some of coins.

Follow the above approach and find the which coin could be needed and how many times?

**Code:** #include<bits/stdc++.h>

using namespace std;

void coin\_change(int coins[], int n, int m)

{

int cnt[n],i;

memset(cnt,0,sizeof cnt);

for(i=n-1;i>=0;i--)

{

if(coins[i]<=m)

{

cnt[i]=m/coins[i];

m=m%coins[i];

}

}

if(m!=0)

cout<<"Change is not possible"<<endl;

else

{

cout<<"Coin need:"<<endl;

for(i=n-1;i>=0;i--)

{

if(cnt[i]!=0)

cout<<coins[i]<<" coin "<<cnt[i]<<" times"<<endl;

}

}

}

int main()

{

int n=5,change=12;

int coins[]={2,5,3,4,6};

sort(coins, coins+n);

coin\_change(coins,n,change);

return 0;

}

**Fibonacci 01(Dynamic):**

**Q:**Write a program to display nth term of Fibonacci series of n terms.

**Code:** #include<bits/stdc++.h>

using namespace std;

int fib(int n)

{

int f[n+2],i;

f[0] = 0;

f[1] = 1;

for (i = 2; i <= n; i++)

{

f[i] = f[i-1] + f[i-2];

}

return f[n];

}

int main()

{

int n;

cout<<"Enter Any Number:";

cin>>n;

cout<<"Fibonacchi Number: "<<fib(n)<<endl;

return 0;

}

**Fibonacci 02(Dynamic):**

**Q:** Are you know the Fibonacci series? If yes, given a positive value with t. Then find the solution & also print it.

**Code:** #include<bits/stdc++.h>

using namespace std;

int fib(int n)

{

int f[n+2],i;

f[0] = 0;

f[1] = 1;

for (i = 2; i <= n; i++)

{

f[i] = f[i-1] + f[i-2];

}

return f[n];

}

int main()

{

int n,t;

cout<<"Test Case:";

cin>>t;

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

cout<<"Number "<<i<<": ";

cin>>n;

cout<<"Fibonacchi "<<i<<" :"<<fib(n)<<endl;

}

return 0;

}