**Question 1**

Template<type> T

T max(T a, T b, T c){

T max = a;

If (b > max) { max = b;}

If (c > max) { max = c;}

}

**Question 2**

#include <iostream>

#include <random>

#include <string>

int main(){

constant std::string type[4] ={

‘H’,

‘D’,

‘S’,

‘C’

}

constant std::string num[13] ={

‘A’,

‘2’,

‘3’,

‘4’,

‘5’,

‘6’,

‘7’,

‘8’,

‘9’,

‘10’,

‘J’,

‘Q’,

‘K’

}

std::default\_random\_engine generator{static\_cast<unsigned int>(time(0))};

std:: uniform\_int\_distribution<int > distribution\_type{1,4};

std:: uniform\_int\_distribution<int > distribution\_num{1,13};

int type\_sp, num\_sp;

std::string res[5];

for (int i = 1; i<=5; i++){

type\_sp = distribution\_type(generator);

num\_sp = distribution\_type(generator);

bool f = Ture;

std::string temp = type[type\_sp-1]+ num[num\_sp-1]

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

if (temp == res[j]){f=False;}

}

if (f) {std::cout << temp << ‘,’;res[i-1]= temp}

else {i-=1;}

}

}

**Question 3**

#include <iostream>

int cal(int a, int b){

int max = a;

int min = b;

if (b>max){max=b;min=a;}

int sum = 0;

for (int i=min;i<=max;i++){sum+=i;}

return sum;

}

int main(){

std::cout << “Please type in two positive integer:/n”;

std::cin >> a >> b;

int sum = cal(a,b);

std::cout << “The sum of ” << a << “to” << b << “is” << sum << “,”;

std::cout << sum << “/100 is” << std::setprecision(3) << sum/100;

// there are other ways to control output

}

**Question 4**

for (int k=2; k <=20; k+=3) {

if (k == 3) continue;

cout << k <<”, “ << k\*k << endl;

}

Covert:

int k = 2;

while(k<=20){

if (k == 3) continue;

std::cout << k <<”, “ << k\*k << std::endl;

k+=3;

}

**Question 5**

class Account(){

public:

// This is called constructor;

Account(std::string fn, std::string ln, std::string add, std::string ssn, float bal):

Fn{fn},Ln{ln},Add{add},Ssn{ssn},Bal(bal){}

std::string get\_Fn(){

return Fn;

}

void set\_Fn(fn){

Fn = fn;

}

std::string get\_Ln(){

return Ln;

}

void set\_Ln(ln){

Ln = ln;

}

// other getter and setter are similar

void toString(){

std::cout << “First name: ” << Fn << “\n”;

std::cout << “Last name: ” << Ln << “\n”;

std::cout << “Address: ” << Add << “\n”;

std::cout << “SSN: ” << Ssn << “\n”;

std::cout << “Current Balance: ” << Bal << “\n”;

}

private:

std::string Fn;

std::string Ln;

std::string Add;

std::string Ssn;

float Bal;

}

**Question 6**

**Recursive:**

int fac(n){

if (n == 1){

return 1;

}

else{

return fac(n-1)\*n

}

}

**Iterative**

fac = 1;

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

fac = fac\*i;

}