1. **Write a program to check whether a given sentense is an Pangram or not(contains A to Z each letter atleast once).**

**public** **class** PangramChecker {

**private** **static** **final** **int** *NO\_OF\_LETTERS\_OF\_ALPHABET* = 26;

**public** **static** **void** main(String[] args) {

String sentence = "Freight to me sixty dozen quart jars and twelve "

+ "black pans.";

System.*out*.println("Is pangram : " + *isPangram*(sentence));

}

**private** **static** **boolean** isPangram(String sentence) {

**if** (sentence.length() < *NO\_OF\_LETTERS\_OF\_ALPHABET*) {

**return** **false**;

}

**for** (**char** ch = 'A'; ch <= 'Z'; ch++) {

System.*out*.println(ch);

System.*out*.println((**char**)(ch+32));

**if** (sentence.indexOf(ch) < 0

&& sentence.indexOf((**char**) (ch + 32)) < 0) {

**return** **false**;

}

}

**return** **true**;

}

}

1. **Write a program to prove that strings are immutable.**

public final class Contacts {

private final String name;

private final String mobile;

public Contacts(String name, String mobile) {

this.name = name;

this.mobile = mobile;

}

public String getName(){

return name;

}

public String getMobile(){

return mobile;

}

}

1. **Given array of n integers and given a number X, find all the unique pairs of elemens (a,b), whose summation is equal to X.**

int main(void)

{

int arr [10] = {1,2,3,4,5,6,7,8,9,0};

findpair(arr, 10, 7);

}

void findpair(int arr[], int len, int sum)

{

std::sort(arr, arr+len);

int i = 0;

int j = len -1;

while( i < j){

while((arr[i] + arr[j]) <= sum && i < j)

{

if((arr[i] + arr[j]) == sum)

cout << "(" << arr[i] << "," << arr[j] << ")" << endl;

i++;

}

j--;

while((arr[i] + arr[j]) >= sum && i < j)

{

if((arr[i] + arr[j]) == sum)

cout << "(" << arr[i] << "," << arr[j] << ")" << endl;

j--;

}

}

}