Mendiyarov Dias SE - 2318

Problem 1

#include <iostream>

#include <climits>

using namespace std;

int findMinimum(int n, int arr[]) {

int minimum = INT\_MAX;

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

if (arr[i] < minimum) {

minimum = arr[i];

}

}

return minimum;

}

int main() {

int arr[] = { 5, 3, 9, 1, 7 };

int size = sizeof(arr) / sizeof(arr[0]);

cout << "Minimum: " << findMinimum(size, arr) << endl;

return 0;

}

Problem 2

#include <iostream>

using namespace std;

double findAverage(int n, int arr[]) {

double sum = 0;

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

sum += arr[i];

}

return sum / n;

}

int main() {

int arr[] = { 5, 3, 9, 1, 7 };

int size = sizeof(arr) / sizeof(arr[0]);

cout << "Average: " << findAverage(size, arr) << endl;

return 0;

}

Problem 3

#include <iostream>

using namespace std;

bool isPrime(int n) {

if (n <= 1) {

return false;

}

for (int i = 2; i \* i <= n; ++i) {

if (n % i == 0) {

return false;

}

}

return true;

}

int main() {

int num = 17;

cout << num << " is " << (isPrime(num) ? "" : "not ") << "prime" << endl;

return 0;

}

Problem 4

#include <iostream>

using namespace std;

unsigned long factorial(int n) {

if (n == 0 || n == 1) {

return 1;

}

return n \* factorial(n - 1);

}

int main() {

int num = 5;

cout << "Factorial of " << num << ": " << factorial(num) << endl;

return 0;

}

Problem 5

#include <iostream>

int fibonacci(int n) {

if (n <= 0) {

return 0;

}

else if (n == 1) {

return 1;

}

else {

return fibonacci(n - 1) + fibonacci(n - 2);

}

}

int main() {

int n = 5;

std::cout << "The " << n << "th Fibonacci number is: " <<

fibonacci(n) << std::endl;

return 0;

}

Problem 6

#include <iostream>

using namespace std;

double power(double base, int exponent) {

if (exponent == 0) {

return 1;

}

return base \* power(base, exponent - 1);

}

int main() {

double base = 2.5;

int exponent = 3;

cout << base << " raised to the power " << exponent << " is " << power(base, exponent) << endl;

return 0;

}

Problem 7

#include <iostream>

#include <algorithm>

using namespace std;

void permute(string str) {

sort(str.begin(), str.end());

do {

cout << str << endl;

} while (next\_permutation(str.begin(), str.end()));

}

int main() {

string str = "ABC";

cout << "Permutations of " << str << " are:" << endl;

permute(str);

return 0;

}

Problem 8

#include <iostream>

using namespace std;

bool isAllDigits(string s) {

for (char c : s) {

if (!isdigit(c)) {

return false;

}

}

return true;

}

int main() {

string testString = "1234";

cout << testString << " consists of " << (isAllDigits(testString) ? "" : "not ") << "digits only" << endl;

return 0;

}

Problem 9

#include <iostream>

int binomialCoeff(int n, int k) {

int c = 1;

if (k > n - k)

k = n - k;

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

c \*= (n - i);

c /= (i + 1);

}

return c;

}

int main() {

int n = 7, k = 3;

std::cout << "Binomial Coefficient of " << n << " and " << k << " is: " << binomialCoeff(n, k) << std::endl;

return 0;

}

Problem 10

#include <iostream>

using namespace std;

int gcd(int a, int b) {

if (b == 0) {

return a;

}

return gcd(b, a % b);

}

int main() {

int a = 84, b = 18;

cout << "GCD of " << a << " and " << b << " is: " << gcd(a, b) << endl;

return 0;

}