1. *#include* <iostream>
2. *#include* <vector>
3. *using* *namespace* std;
4. *int* main() {
5. *int* n\_alunos;
6. cout *<<* "Insira o número de alunos: ";
7. cin *>>* n\_alunos;
8. vector*<int>* notas(n\_alunos);
9. vector*<int>* histograma(5, 0);
10. *for* (*int* i *=* 0; i *<* n\_alunos; i*++*) {
11. cout *<<* "Insira a nota " *<<* i*+*1 *<<* ": ";
12. cin *>>* notas*[*i*]*;
13. *int* indice *=* (notas*[*i*]* *>=* 9) *?* 0 *:*
14. (notas*[*i*]* *>=* 7) *?* 1 *:*
15. (notas*[*i*]* *>=* 5) *?* 2 *:*
16. (notas*[*i*]* *>=* 3) *?* 3 *:* 4;
17. histograma*[*indice*]++*;
18. }
19. *char* conceitos[5] *=* {'A', 'B', 'C', 'D', 'E'};
20. *for* (*int* i *=* 0; i *<* 5; i*++*) {
21. cout *<<* conceitos[i] *<<* ": ";
22. *for* (*int* j *=* 0; j *<* histograma*[*i*]*; j*++*) {
23. cout *<<* "\*";
24. }
25. cout *<<* endl;
26. }
27. *return* 0;
28. }

2.

*#include* <iostream>

*#include* <cmath>

*using* *namespace* std;

*class* equacaoSegundoGrau {

*private:*

*double* a, b, c;

*public:*

equacaoSegundoGrau (*double* a, *double* b, *double* c) : a(a), b(b), c(c) {}

*double* calcularDelta() {

*return* pow(b, 2) *-* 4 *\** a *\** c;

}

*void* exibirCoeficientes() {

cout *<<* "Coeficientes: a = " *<<* a *<<* ", b = " *<<* b *<<* ", c = " *<<* c *<<* endl;

}

};

*int* main() {

*double* a, b, c;

cout *<<* endl;

cout *<<* "Cálculo de equações do segundo grau em C++" *<<* endl;

cout *<<* endl;

cout *<<* "Digite o coeficiente A: " *<<* endl;

cin *>>* a;

cout *<<* "Digite o coeficiente B: " *<<* endl;

cin *>>* b;

cout *<<* "Digite o coeficiente C: " *<<* endl;

cin *>>* c;

equacaoSegundoGrau equacao(a, b, c);

equacao.exibirCoeficientes();

*double* delta *=* equacao.calcularDelta();

cout *<<* "Δ = " *<<* delta *<<* endl;

*return* 0;

}

3.

*#include* <iostream>

*#include* <vector>

*using* *namespace* std;

*class* vetorNumerosInteiros {

*private:*

vector*<int>* vetor;

*public:*

vetorNumerosInteiros(*int* *tamanho*) : vetor(*tamanho*) {}

*void* getValores() {

cout *<<* "Digite os valores do vetor: \n";

*for* (*int* i *=* 0; i *<* vetor.size(); i*++*) {

cout *<<* "Valor " *<<* i *+* 1 *<<* ": ";

cin *>>* vetor*[*i*]*;

}

}

*void* doubleValores() {

*for* (*int* i *=* 0; i *<* vetor.size(); i*++*) {

vetor*[*i*]* *\*=* 2;

}

}

*void* showValores() *const* {

cout *<<* "Valores dentro do vetor: \n";

*for* (*int* i *=* 0; i *<* vetor.size(); i*++*) {

cout *<<* vetor*[*i*]* *<<* " ";

}

cout *<<* endl;

}

};

*int* main() {

*int* tamanhoVetor;

cout *<<* "Informe a quantidade de itens do vetor: ";

cin *>>* tamanhoVetor;

vetorNumerosInteiros vetorUsuario(tamanhoVetor);

vetorUsuario.getValores();

vetorUsuario.doubleValores();

vetorUsuario.showValores();

*return* 0;

}

4.

*#include* <iostream>

*using* *namespace* std;

*int* recursiveSum(*int* *a*, *int* *b*) {

*if* (a *==* b) {

*return* a;

}

*return* a *+* recursiveSum(a *+* 1, b);

}

*int* main () {

*int* a,b;

cout *<<* "Inisira o valor de A: " *<<* endl;

cin *>>* a;

cout *<<* "Inisira o valor de B: " *<<* endl;

cin *>>* b;

*if* (a *<* b) {

*int* soma *=* recursiveSum(a, b);

cout *<<* "Soma = " *<<* soma *<<* endl;

} *else* {

cout *<<* "Operação inválida. Insira um valor onde B seja maior que a A." *<<* endl;

}

}

5.

*#include* <iostream>

*#include* <cstdlib>

*#include* <ctime>

*#include* <algorithm>

*using* *namespace* std;

*int* bbr(*int* *v*[], *int* *baixo*, *int* *alto*, *int* *chave*) {

*if* (*baixo* *>* *alto*) {

*return* *-*1;

}

*int* meio *=* (*baixo* *+* *alto*) */* 2;

*if* (*v*[meio] *==* *chave*) {

*return* meio;

}

*if* (*v*[meio] *<* *chave*) {

*return* bbr(*v*, meio *+* 1, *alto*, *chave*);

} *else* {

*return* bbr(*v*, *baixo*, meio *-* 1, *chave*);

}

}

*int* main() {

*const* *int* size *=* 50;

*int* v[size];

srand(time(nullptr));

*for* (*int* i *=* 0; i *<* size; i*++*) {

v[i] *=* rand() *%* 101;

}

sort(v, v *+* size);

cout *<<* "Vetor ordenado: ";

*for* (*int* i *=* 0; i *<* size; i*++*) {

cout *<<* v[i] *<<* " ";

}

cout *<<* endl;

*int* chave;

cout *<<* "Digite um valor para realizar a busca binária recursiva: ";

cin *>>* chave;

*int* resultado *=* bbr(v, 0, size *-* 1, chave);

*if* (resultado *!=* *-*1) {

cout *<<* "O valor " *<<* chave *<<* " foi encontrado no índice " *<<* resultado *<<* endl;

} *else* {

cout *<<* "O valor " *<<* chave *<<* " não foi encontrado no vetor." *<<* endl;

}

*return* 0;

}

6.