BÀI TẬP 1CHUYÊN ĐỀ TỔ CHỨC DỮ LIỆUKÌ 2 2022-2023, HỆ ĐÀO TẠO TỪ XA

MSSV: 21880159

Họ và Tên: Nguyễn Hữu Vinh

1. (2 đ). Bài tập 1.5.7.  


Mã nguồn C++:

|  |
| --- |
| #include <iostream>  using namespace std;  int sumOfDegits(unsigned long long n) {  int sum = 0;  while (n != 0) {  sum = sum + n % 10;  n = n / 10;  }  return sum;  } |

Chạy thử với n = 123456789123456789:

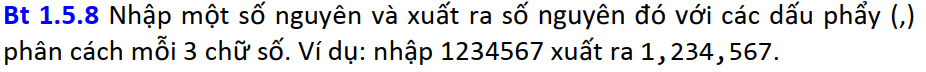
|  |
| --- |
| int main(){  unsigned long long n = 123456789123456789;  cout << sumOfDegits(n);  return 0;  } |

Kết quả chạy thử:

Graphical user interface, text

Description automatically generated

**2. (2 đ). Bài tập 1.5.8.**



Mã nguồn C++:

|  |
| --- |
| #include <iostream>  #include <cstdlib>  using namespace std;  int length(int n) {  int len = 1;  while (n >= 10)  {  n /= 10;  len++;  }  return len;  }  int main()  {  int j, i, k = 0, n;  cout << "Nhap mot so nguyen: ";  cin >> n;  cout << "Ket qua dinh dang: ";  //neu la so am in dau -  if (n < 0)  {  cout << '-';  n \*= -1;  }  int temp = n;  i = length(n);  j = i % 3;  int p = pow(10, i - 1);  //hien thi ket qua  while (i > 0)  {  cout << n / p;  n %= p;  p /= 10;  i--;  k++;  j--;  if ((k % 3 == 0 && i > 0) || (j == 0 && i > 2))  {  cout << ",";  k = 0;  }  }  return 0;  } |

Chạy thử với input là 123456789:

Text

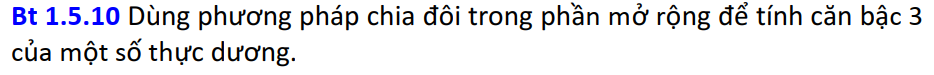
Description automatically generated

Chạy thử với input là -123456789:

Graphical user interface, text, application, chat or text message

Description automatically generated

**3. (2 đ). Bài tập 1.5.10.**



Mã nguồn C++:

|  |
| --- |
| #include <iostream>  #include <math.h>  using namespace std;  double canBacBa(double n)  {  double start = 0, end = n;  double e = 0.0000001;  while (true)  {  double mid = (start + end) / 2;  double error = abs(n - mid \* mid \* mid);  if (error <= e) {  return mid;  }  if ((mid \* mid \* mid) > n) {  end = mid;  }  else {  start = mid;  }  }  }  int main()  {  int dau = 1;  cout << "Nhap mot so thuc: ";  double n = 0;  cin >> n;  if (n < 0) {  dau = -1;  n = -n;  }  cout << "Can bac 3 cua " << n \* dau << " la " << canBacBa(n) \* dau;  return 0;  } |

Chạy thử với -7:

Text

Description automatically generated

Chạy thử với 7:

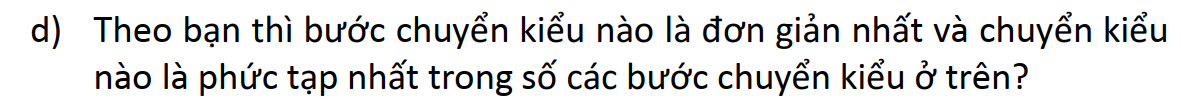
Text

Description automatically generated

**5. (2 đ). Bài tập 3.5.4.**

Text

Description automatically generated



1. Giá trị của:

* a = 1
* b = 60.3
* c = 1

1. Dòng //1:

+ ‘1’(char) -> 49 (int) => 49 > 1 : true (bool) -> 1 (int)

+ 1(int) -> 1.0 (double) => 1.0 < 1.0 : false (bool) -> 0 (int)

* a = 1 + 0 = 1(int);

Dòng //2:

+ ‘2’(char) -> 50(int) -> 50.0(double)

* b = 10.3 + 50.0 = 60.3(double)

Dòng //3:

+ (int)1.5 -> 1(int) -> 1.0(double)

* c = 1.0 + 0.5 = 1.5 -> 1(int)

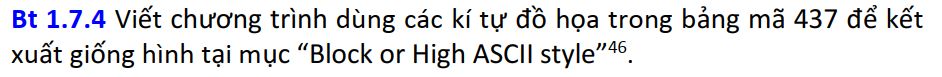
1. Ép kiểu tường minh xảy ra ở dòng //3 cụ thể là đoạn mã *(int)1.5*. Nếu không có ép kiểu tường minh kết quả sẽ thay đổi:

int c = 1.5 + 0.5 = 2.0 -> 2(int)

Ban đầu kết quả có ép kiểu tường minh thì c = 1 nhưng không ép kiểu tường minh thì c = 2.

1. Theo em, bước chuyển kiểu ở dòng //2 phức tạp nhất vì phải chuyển từ ‘2’ có kiểu char -> int -> double.

Bước chuyển kiểu tường minh ở dòng //3 (int)1.5 là dễ hiểu rõ, ràng nhất.

**4. (2 đ). Bài tập 1.7.4.** Mã nguồn C++:

|  |
| --- |
| #include <iostream>  using namespace std;  string spaces(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += " ";  }  return temp;  }  string upperBlock(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(223);  }  return temp;  }  string lowerBlock(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(220);  }  return temp;  }  string highBlackBlock(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(219);  }  return temp;  }  string highScratchtBlock1(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(178);  }  return temp;  }  string highScratchtBlock2(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(177);  }  return temp;  }  string highScratchtBlock3(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(176);  }  return temp;  }  string highHalfBlackBlock(int len) {  string temp = "";  for (int i = 0; i < len; i++)  {  temp += char(221);  }  return temp;  }  int main() {  cout << spaces(5) << lowerBlock(1) << spaces(13)  << lowerBlock(1) << highBlackBlock(1)  << lowerBlock(1) << spaces(1) << highBlackBlock(1)  << lowerBlock(1) << spaces(7)  << lowerBlock(1) << endl;  cout << spaces(1) << lowerBlock(1) << highBlackBlock(1)  << upperBlock(1) << highBlackBlock(1)  << highScratchtBlock1(1) << spaces(1)  << lowerBlock(1) << highScratchtBlock1(1)  << upperBlock(2) << highBlackBlock(1)  << upperBlock(1) << spaces(1) << upperBlock(3)  << highBlackBlock(1) << highScratchtBlock1(1)  << upperBlock(2) << spaces(1)  << upperBlock(2) << spaces(4) << lowerBlock(1)  << highBlackBlock(1) << upperBlock(1)  << highBlackBlock(1) << highScratchtBlock1(1)  << upperBlock(5) << highScratchtBlock1(1)  << lowerBlock(1) << upperBlock(1) << highBlackBlock(2)  << upperBlock(2) << endl;  cout << highBlackBlock(2) << spaces(2) << highBlackBlock(2)  << spaces(1) << upperBlock(1)  << highBlackBlock(2) << lowerBlock(2) << spaces(1)  << lowerBlock(1) << highBlackBlock(1)  << spaces(2) << upperBlock(1) << spaces(1)  << highScratchtBlock3(1) << highScratchtBlock2(1)  << spaces(1) << highScratchtBlock3(1)  << highScratchtBlock2(1) << spaces(3) << highBlackBlock(2)  << spaces(2) << highBlackBlock(2) << spaces(1)  << lowerBlock(1) << highBlackBlock(1)  << lowerBlock(1) << spaces(1) << highBlackBlock(1)  << upperBlock(1) << spaces(1)  << highBlackBlock(2) << endl;  cout << highBlackBlock(1) << highScratchtBlock1(1)  << lowerBlock(1) << upperBlock(1)  << highBlackBlock(2) << spaces(2) << lowerBlock(1)  << spaces(1) << upperBlock(1)  << highBlackBlock(1) << highHalfBlackBlock(1)  << highScratchtBlock1(1) << highBlackBlock(1)  << spaces(4) << highScratchtBlock2(1)  << highScratchtBlock1(1) << spaces(1)  << highScratchtBlock2(1)  << highScratchtBlock1(1) << spaces(3)  << highBlackBlock(1) << highScratchtBlock1(1)  << lowerBlock(1)  << upperBlock(1) << highBlackBlock(2) << spaces(1)  << highScratchtBlock1(1) << highBlackBlock(1)  << spaces(1) << upperBlock(1) << lowerBlock(1)  << spaces(2) << highBlackBlock(1)  << highScratchtBlock1(1) << endl;  cout << highBlackBlock(1) << highScratchtBlock2(1) << spaces(2)  << highBlackBlock(1) << highScratchtBlock1(1)  << spaces(1) << highBlackBlock(2) << lowerBlock(1)  << highScratchtBlock1(1) << upperBlock(1)  << spaces(1) << upperBlock(1) << highBlackBlock(1)  << lowerBlock(2) << highBlackBlock(1)  << lowerBlock(1) << highScratchtBlock1(1)  << highBlackBlock(1) << spaces(1) << highScratchtBlock1(1)  << highBlackBlock(1) << spaces(3) << highBlackBlock(1)  << highScratchtBlock2(1) << spaces(2)  << highBlackBlock(1) << highScratchtBlock1(1) << spaces(1)  << highScratchtBlock2(1) << highBlackBlock(1)  << spaces(2) << highScratchtBlock1(1)  << highBlackBlock(1) << lowerBlock(1) << spaces(1)  << highScratchtBlock2(1) << endl;  cout << spaces(4) << upperBlock(1) << highScratchtBlock2(1)  << spaces(11) << upperBlock(1) << spaces(2)  << upperBlock(1) << spaces(1) << highBlackBlock(1)  << upperBlock(1) << spaces(7) << upperBlock(1)  << highScratchtBlock2(1) << spaces(2) << upperBlock(1)  << spaces(2) << highBlackBlock(1) << upperBlock(1)  << spaces(2) << highScratchtBlock3(1) << endl;  } |

Kết quả sau khi chạy chương trình:

(Font chữ sử dụng cho Console window là **Cascadia Code**)

Qr code

Description automatically generated