深蓝学院
shenlanxueyuan.com

东北陈小春第二次作业分享

主讲人 东北陈小春

纲要

深蓝学院
shenlanxueyuan.com

- 第一部分：框架
- 第二部分：流程介绍
- 第三部分：函数实现

框架

深蓝学院
shenlanxueyuan.com

- 加减号符号定义

```
constexpr char sign_sub = '-';
constexpr char sign_add = '+';
```

- 函数定义

```
/* 把字符串每个字符解析成数字*/
int convert_char_to_int(char str_){…
/* 把数字转换成字符存储*/
char convert_int_to_char(int val){…
```

框架

深蓝学院
shenlanxueyuan.com

- 函数定义

```
/*加法函数，正正情况和负负情况*/
std::string fun_add1(std::string str1, std::string str2, int base){…

/*减法函数，正负情况和负正情况*/
std::string fun_sub1(std::string str1, std::string str2, int base){…

/*输出进制转换函数*/
std::string conrt_base(std::string result1, int m_base, int n_base){…
```

```
int main()…
```

纲要

深蓝学院
shenlanxueyuan.com

- 第一部分：概述
- **第二部分：流程介绍**
- 第三部分：函数内容介绍

流程介绍

深蓝学院 shenlanxueyuan.com

Number_base,out_base,lnumber1和lnumber采用相同的形式while判断

```
std::string lnumber1,lnumber2;      //两个长整数和存放结果的字符串
std::string result;
int number_base;     //输入进制位数
int out_base;     //输出进制位数
std::cout << "请确定输入的两个长整数进制数" << std::endl;
while (!(std::cin >> number_base) || number_base <= 0 || number_base > 32)
{
    std::cin.clear();
    fflush(stdin);
    std::cout << "输入有误，请重新确定输入进制数";
}
```

流程介绍

深蓝学院 shenlanxueyuan.com

对符号进行处理

```
if(*lnumber1.cbegin() == sign_sub && *lnumber2.cbegin() == sign_sub){
    lnumber1.erase(0,1);
    lnumber2.erase(0,1);
    result = '-' + fun_add1(lnumber1,lnumber2,number_base_);
```

“+ -”和“- +”情况需判断大小，大数减小数以及额外处理

```
int lnum1begin = convert_char_to_int(*lnumber1.cbegin());
int lnum2begin = convert_char_to_int(*lnumber2.cbegin());
bool sign_ = lnumber1.size()>lnumber2.size() ||
(lnumber1.size() == lnumber2.size() && ((lnumber1.compare(lnumber2) == 1)));
result = fun_sub1(lnumber1,lnumber2,number_base_);
```

流程介绍

深蓝学院
shenlanxueyuan.com

```
/*判断输入为正数还是负数，并且判断是否含有正负号，有的话去除正负号*/
if(*lnumber1.cbegin() == sign_sub && *lnumber2.cbegin() == sign_sub){…
else if(*lnumber1.cbegin() == sign_sub && *lnumber2.cbegin() != sign_sub){…
else if(*lnumber1.cbegin() != sign_sub && *lnumber2.cbegin() == sign_sub){…
else{…
```

```
//转换成相应的进制
//number_base -> out_base
    result = conrt_base(result,number_base,out_base);

//打印结果
    std::cout << "计算的结果是: "  << result << std::endl;
}
```

流程介绍

深蓝学院
shenlanxueyuan.com

结果等于“00012”删0处理

```
//去掉差前面的0
for(; i < result.length(); ++i){
    if(result[i]!='0')break;
}
result = result.substr(i);
```

结果为负需加上“-”

```
if(sign_ ){
    result = sign_sub + result;
}
```

纲要

深蓝学院
shenlanxueyuan.com

- 第一部分：概述
- 第二部分：流程介绍
- **第三部分：函数内容介绍**

函数 “+ +”和“- -”1

深蓝学院 shenlanxueyuan.com

```
std::string fun_add1(std::string str1, std::string str2, int base){
for (int i = 0; i < len; ++i) {
    char str1_i = i < str1.size() ? str1[i] : '0';
    char str2_i = i < str2.size() ? str2[i] : '0';
    int intstr1_i = convert_char_to_int(str1_i);
    int intstr2_i = convert_char_to_int(str2_i);
    int val = (intstr1_i + intstr2_i + carry) % base;
    char val2 = convert_int_to_char(val);
    carry = (intstr1_i + intstr2_i + carry) / base;
    {
        std::ostringstream obj1;
        obj1 << val2;
        obj1 << result1;
        result1 = obj1.str();
    }
}
```

对两个字符串翻转，每位计算结果存入reverse1

函数 “+ –”和“– +”1

深蓝学院
shenlanxueyuan.com

```
std::string fun_sub1(std::string str1, std::string str2, int base){
```

Number1和number2绝对值大小判断

```
int len = std::max(str1.size(), str2.size());
/*绝对值大的数做str1*/
bool sign_size = (str1.size()>str2.size() || (str1.size() == str2.size() && (str1.compare(str2) == 1)));
if(sign_size == false){
    std::string tmp = str1;
    str1 = str2;
    str2 = tmp;
}
```

函数 “+ −”和“− +”2

深蓝学院
shenlanxueyuan.com

```
for (int i = 0; i < len; ++i) {
    char str1_i = i < str1.size() ? str1[i] : '0';
    char str2_i = i < str2.size() ? str2[i] : '0';
    int intstr1_i = convert_char_to_int(str1_i);
    int intstr2_i = convert_char_to_int(str2_i);
    int val_ = intstr1_i - intstr2_i - carry_;
    carry_ = 0;                                    //使用完后借位运算符置零
    /*判断是否需要借位*/
    if(val_ < 0){
        val_ += base;
        carry_ = 1;
    }
    char val2 = convert_int_to_char(val_);
    {
        std::ostringstream obj2;
        obj2 << val2;
        obj2 << result1;
        result1 = obj2.str();
    }
}
```

字符数字转换函数

深蓝学院
shenlanxueyuan.com

```
int convert_char_to_int(char str_){
    int a = str_ - '0';
    if(a >= 49 && a <= 74){
        a -= 39;          //小写字母处理
    }
    else if(a >= 17 && a <= 42){
        a -= 7;           //大写字母处理
    }
    else{
        //数字不处理
    }
    return a;
}
```

对于结果的 进制转换函数1

深蓝学院 shenlanxueyuan.com

```
std::string conrt_base(std::string result1, int m_base, int n_base){
```

```
    //去除符号位
    bool sub_ = (*result1.cbegin() == '-');
    if(sub_){
        result1.erase(0,1);
    }
```

变量定义

```
std::string result2 = "";     //存放转换后的结果串
int currentnum = 0;           //当前位
int mod = 0;                  //余数
std::string quotient = result1;      //被除数商串
std::string temp_quotient="";        //存放中间商串
```

进制转换函数2

深蓝学院
shenlanxueyuan.com

从字符串最高位开始除输出进制数，余数乘当前进制数加下一位继续除输出进制数得到最终余数，余数压入结果串

```
for(int  i= 0;i < quotient.length(); ++i){      //该循环即从
    currentnum = convert_char_to_int(quotient[i]);
    mod = mod * m_base + currentnum;
    temp_quotient += convert_int_to_char(mod/n_base);
    mod = mod % n_base;
}
```

```
result2 = convert_int_to_char(mod) + result2;
```

进制转换函数3

深蓝学院
shenlanxueyuan.com

更新被除数，循环上一步操作，每次压入一个余数，得到最终结果

```
  quotient = temp_quotient;
//初到被除商为0
while(quotient.length()>0){
```

删除前面的0

```
for(; i < quotient.length(); ++i){          //去掉商前面的0
    if(quotient[i]!='0')break;
}
quotient=quotient.substr(i);
```

结果补上符号位

```
  /*   如果是负数补上负号     */
  if(sub_){
      result2 = sign_sub + result2;
  }
```

深蓝学院
shenlanxueyuan.com

感谢各位聆听！

Thanks for Listening