memset 优化实验

2023 年 1 月 3 日

班　　级: 计试 001

姓　　名: 唐梓航

学　　号: 2206113908

目录

1 时间测量

为了在之后的优化过程中排除测量方式等因素对时间测量以及优化好坏判断的影响，对时间测量方式进行探究是有意义的。经过实验，第一次运行函数时速度总会很慢。并且，该现象的产生是

```
void *basic_memset(void *s,int c,size_t n){
        size_t cnt=0;
        unsigned char *tmp=(unsigned char*) s;
        while(cnt<n) *tmp++=(unsigned char) c,cnt++;
        return s;
}

int target[100000000];

int main(){

        for(int i=1;i<=10;i++){
                double tmp=clock();
                memset(target,i,sizeof(target));
                cout<<(clock()-tmp)/(double)CLOCKS_PER_SEC<<endl;
        }
        return 0;
}
☁  memset [master] ⚡  ./a.out
0.132138
0.017386
0.017726
0.016774
0.017608
0.01688
0.017653
0.017584
0.017148
0.017282
```

多种因素共同导致的，很难避免。为了更好的模拟实际情况，在后续的实验中，会对优化后的时间均进行统计，但会以第一次为主要依据。

2 循环展开

通过循环展开的方式进行优化，成功达到了一定的优化效果

```
0.130222
0.017035
0.01753
0.016353
0.0174
0.017068
0.016837
0.016999
0.017293
0.017334
☁  memset [master] ⚡  cat xunhuan.cpp
#include<bits/stdc++.h>

using namespace std;


void *basic_memset(void *s,int c,size_t n){
        size_t cnt=0;
        unsigned char *tmp=(unsigned char*) s;
        while(cnt+4<n){
                *(tmp)=c;
                *(tmp+1)=c;
                *(tmp+2)=c;
                *(tmp+3)=c;
                tmp+=4;
        }
        //while(cnt<n) *tmp++=(unsigned char) c,cnt
        return s;
}
```

3 32 位操作

由于机器字长为 32 位，因此通过以 32 位为一个单位的操作达到了更好的优化效果。

```
☁  memset [master] ⚡  ./a.out
0.129174
0.016857
0.017649
0.017439
0.016811
0.017231
0.016789
0.017284
0.016705
0.016858
☁  memset [master] ⚡  cat int.cpp
#include<bits/stdc++.h>

using namespace std;

int target[100000000];

void *basic_memset(void *s,int c,size_t n){
        size_t cnt=0;
        unsigned int cc=0;
        for(int i=1;i<=4;i++) cc<<=8,cc+=(unsigned
        unsigned int *tt=(unsigned int *)s;
        int nn=n/4-1;
        while(cnt<nn) *tt++=cc,cnt++;
        cnt*=4;
        unsigned char *tmp=(unsigned char*) tt;
        while(cnt<n) *tmp++=c,cnt++;
        return s;
}
```