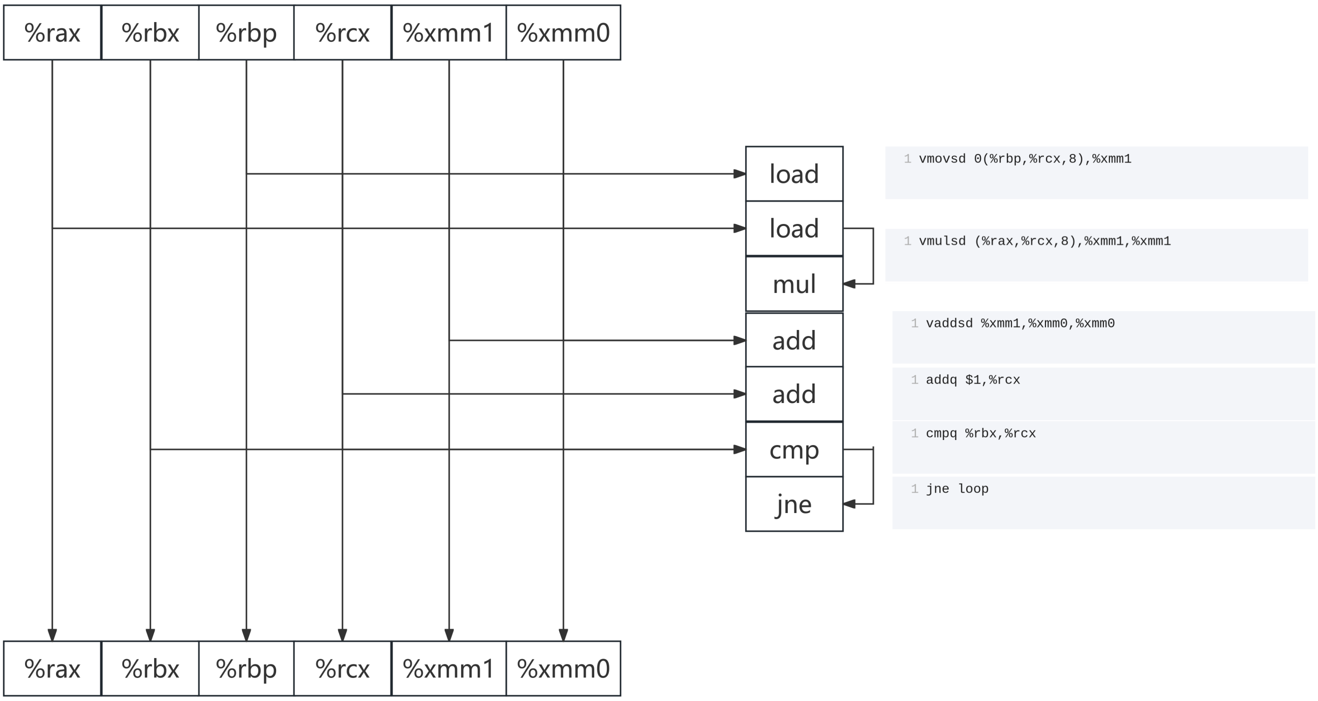
2025计算机系统软件基础-第3次作业-张睿恒-2024302182001

$T_{5.13}$

A:如图所示



```
vmovsd 0(%rbp,%rcx,8),%xmm1
```

```
vmulsd (%rax,%rcx,8),%xmm1,%xmm1
```

```
vaddsd %xmm1,%xmm0,%xmm0
```

```
addq $1,%rcx
```

```
cmpq %rbx,%rcx
```

```
jne loop
```

B:浮点数加法

C:整数加法

D:虽说浮点数乘法需要5个时钟周期，但是并没有数据依赖，可以并行处理。而执行加法操作需要依赖上次加法操作得到的值，所以加法必须排队。因此，制约程序性能的关键路径在于加法操作而不是浮点数乘法。浮点数加法的CPE为3.00，所以整个函数的CPE在3.00

$T_{5.16}$

```c
void inner4_6plus1a(vec_ptr u,vec_prt v, data_t* dest){
    long i;
    long length=vec_length(u);
    data_t *udata=get_vec_start(u);
    data_t *vdata=get_vec_start(v);
    data_t sum=(data_t)0;
    data_t sum1=(data_t)0;
    data_t sum2=(data_t)0;
    data_t sum3=(data_t)0;
    data_t sum4=(data_t)0;
    data_t sum5=(data_t)0;

    for(i=1;i<length;++i){
        sum1+=udata[i]*vdata[i];
        sum2+=udata[i+1]*vdata[i+1];
        sum3+=udata[i+2]*vdata[i+2];
        sum4+=udata[i+3]*vdata[i+3];
        sum5+=udata[i+4]*vdata[i+4];
        sum=sum+udata[i+5]*vdata[i+5]+sum1+sum2+sum3+sum4+sum5;
    }
    while(i<length){
        sum+=udata[i]*vdata[i];
        ++i;
    }
    *dest=sum;
    return;
}
```

$T_{5.17}$

```c
void *word_memset(void *s, int c, size_t n){
  if (n < K){
      size_t cnt = 0;
      unsigned char *schar = s;
      while (cnt < n){
          *schar++ = (unsigned char)c;
          cnt++;
      }
  }
  else{
      unsigned long word = 0;
      for (int i = 0; i < K; ++i){
          word <<= K*CHAR_BIT;
          word += (unsigned char)c;
      }

      size_t cnt = 0;
      unsigned long *slong = s;
      while (cnt < n){
          *slong++ = word;
          cnt += K;
      }

      unsigned char *schar = slong;
      while (cnt < n){
          *schar++ = (unsigned char)c;
          cnt++;
      }
  }
  return s;
}
```