1. 试讨论冯·诺伊曼结构的主要特点。

(1)查阅资料，分别给出一款Intel、AMD商业处理器的峰值性能和访存带宽。

AMD Ryzen™ 7 3750H

基准时钟频率2.3GHz 最大时钟频率4.0GHz

一级缓存 384KB

二级缓存 2MB

三级缓存 4MB

最高内存速度 2400MHz

Intel Core i7-8750H

基准时钟频率2.2GHz 最大时钟频率4.1GHz

一级缓存 384KB

二级缓存 1.5MB

三级缓存 9MB

最高内存速度 2666MHz

最大内存带宽 41.8 GB/s

(2)分析这2款处理器的访存带宽和存储层次参数（一级cache大小和延迟、二级cache大小和延迟等）之间的关系。

cache大小和延迟成正比关系；cache容量越少，延迟越少

以一台个人计算机上为运行环境：

1. 查阅相关资料，给出该计算机的浮点运算峰值。

英特尔 酷睿i7-8750H 处理器 AVX2指令集

8\*12\*4.1GHz

(2)编写并运行一个1024\*1024双精度矩阵乘法程序，计算出实际浮点运算速度。

#include "stdio.h"

#include "stdlib.h"

#include "time.h"

double a[1024][1024];

double b[1024][1024];

double c[1024][1024];

int main()

{

for (int i = 0; i < 1024; i++) {

for (int j = 0; j < 1024; ++j) {

a[i][j] = rand() / 100.0;

b[i][j] = rand() / 100.0;

}

}

clock\_t start, end;

start = clock();

for (int i = 0; i < 1024; i++) {

for (int j = 0; j < 1024; ++j) {

c[i][j] = 0;

for (int k = 0; k < 1024; ++k)

c[i][j] += a[i][k] \* b[k][j];

}

}

end = clock();

double totaltime = -1.0\*(start - end)/CLK\_TCK;

printf("%lf\n", totaltime);

printf("%lf", (1024\*1024\*1024\*2.0)/ totaltime);

}

单核大约每秒钟3.7Gflops

2.在为了优化处理器的某一部分而进行改变时，经常会出现这样一种情况：加速某种类型的指令时，会降低其他某些指令的速度。例如，如果放入一个复杂的浮点单元，它要占用空间，为了容纳它，就得将某些东西移得远一些，这样就会增加一些延迟周期才能到达被挪远了的单元。基本的Amdahl定律没有考虑这些问题。

a. 如果新增加的某个快速浮点单元使浮点运算平均提高到2倍，浮点运算占用的时间为原程序执行时间的20%，那么总加速比是多少（忽略对所有其他指令的影响）？

1/0.9 = 1.11

b.假定浮点单元的加速会降低数据缓存访问的速度，减缓倍数为1.5倍（即加速比为2/3）。数据缓存访问事件为总执行时间的10%。那么总加速比为多少？

1/(0.7+1.5\*0.1+0.1) = 1/0.95 = 1.05

c.在实现新的浮点运算之后，在浮点运算上花费的执行时间占多大比例？数据缓存访问又占多大比例？

0.1/0.95 = 10.5%

0.15/0.95 = 15.8%