|  |
| --- |
| 题1： |
|  | 1： |
|  | 汇编语言是面向具体CPU及其硬件结构的语言 |
|  | C语言可以做到硬件无关性 |
|  | 2： |
|  | 汇编语言效率较高 |
|  | C语言效率相对较低 |
|  | 3： |
|  | 汇编语言是非结构语言，可读性较弱 |
|  | C语言是结构语言，可读性较强 |
|  | 4： |
|  | 汇编语言基本没有可移植性 |
|  | C言可移植性很强 |
|  | 5： |
|  | 汇编语言没有函数库，所有代码需要自己组织 |
|  | C语言有丰富的函数库可供调用 |
|  | 题2： |
|  | https://github.com/13lim-lin/2019\_Lab1\_os-ai/blob/13lim-lin-patch-1/题2.png |
|  |  |
|  | 题3： https://github.com/13lim-lin/2019\_Lab1\_os-ai/blob/13lim-lin-patch-1/题3.png |
|  | 题4：https://github.com/13lim-lin/2019\_Lab1\_os-ai/blob/13lim-lin-patch-1/题4.PNG |
|  | #include<stdio.h> |
|  | #include<stdlib.h> |
|  |  |
|  | struct Eiglish{ |
|  | char letter; |
|  | struct Eiglish \*next; |
|  |  |
|  | }; |
|  | int main() |
|  | { |
|  | struct Eiglish head,tail,\*p,\*q; |
|  | int i; |
|  | p=(struct Eiglish \*)malloc(sizeof(struct Eiglish)); |
|  | for(i=0;i<26;i++) |
|  | { |
|  | q=(struct Eiglish \*)malloc(sizeof(struct Eiglish)); |
|  | scanf("%c",&p->letter); |
|  | q->next=p; |
|  | p=q; |
|  | head.next=q; |
|  | } |
|  | p=head.next; |
|  | p=p->next; |
|  | for(i=0;i<26;i++) |
|  | { |
|  |  |
|  | printf("%c",p->letter); |
|  | p=p->next; |
|  | } |
|  | } |
|  |  |
|  | } |
|  |  |