COMP2300/6300

Computer Organisation and Programmation

Data structures

Dr Charles Martin

Semester 1, 2022

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

CC BY NC SA

Week 7: Data Structures

- why do we need data structure
- arrays
- records
- memory allocation

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Admin Time

Assignment 1 results

Mid-semester Exam

Psyche up for second half of the course

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Why do we need data structures?

to structure our **data** (um, ok…)

to minimise bookkeeping

to make our lives easier

to play nicely with others

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Arrays

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Arrays are for collections of homogeneous data

程序代写代做 CS编程辅导

Wikipedia has more (as usual)

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

You might need an array if you've got

程序代写代做 CS编程辅导

- a string of ASCII characters (re houty McShoutface?)
- an audio file (e.g. an array of 1 d values to “send” to an audio output)
- a collection of red-green-blue (3 x 8-bit unsigned) values which represent a bitmap image

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Essential information for an array

程序代写代做 CS编程辅导

An array is just a bunch of things. There are the key things to figure out:

- what are the things?
- where are the things?
- how big are the things?
- how many things are there?

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Can you fit an array into a register?

程序代写代做 CS编程辅导

probably not

need to store it in memory

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

程序代写代做 CS编程辅导
WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Addressing

程序代写代做 CS编程辅导

Array addressing is about readin ng specific elements of the array

also called *indexing* (Latin: *indic* t, show")

```
//                the index
//                    ↓
int x = the_things[0]; // the 1st element
int y = the_things[4]; // the 5th element
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

In assembly code

程序代写代做 CS编程辅导

it's just **loads and stores** (like you [illegible] oing all along)

```
   .data
the_things:
   .word 2, 3, 0, 0, 6, 3, 0, 0

   .text
get_elements:
   ldr r0, =the_things
   ldr r1, [r0]      @ the 1st element
   ldr r2, [r0, 16] @ the 5th element
```

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Alignment

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

talk

程序代写代做 CS编程辅导

what's the big deal? why are una ds/stores a problem?

hint: check out the Cortex M4 Tec erence Manual section on **load store timings**

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

程序代写代做 CS编程辅导

you can put *anything* in an array, not just words

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

程序代写代做 CS编程辅导

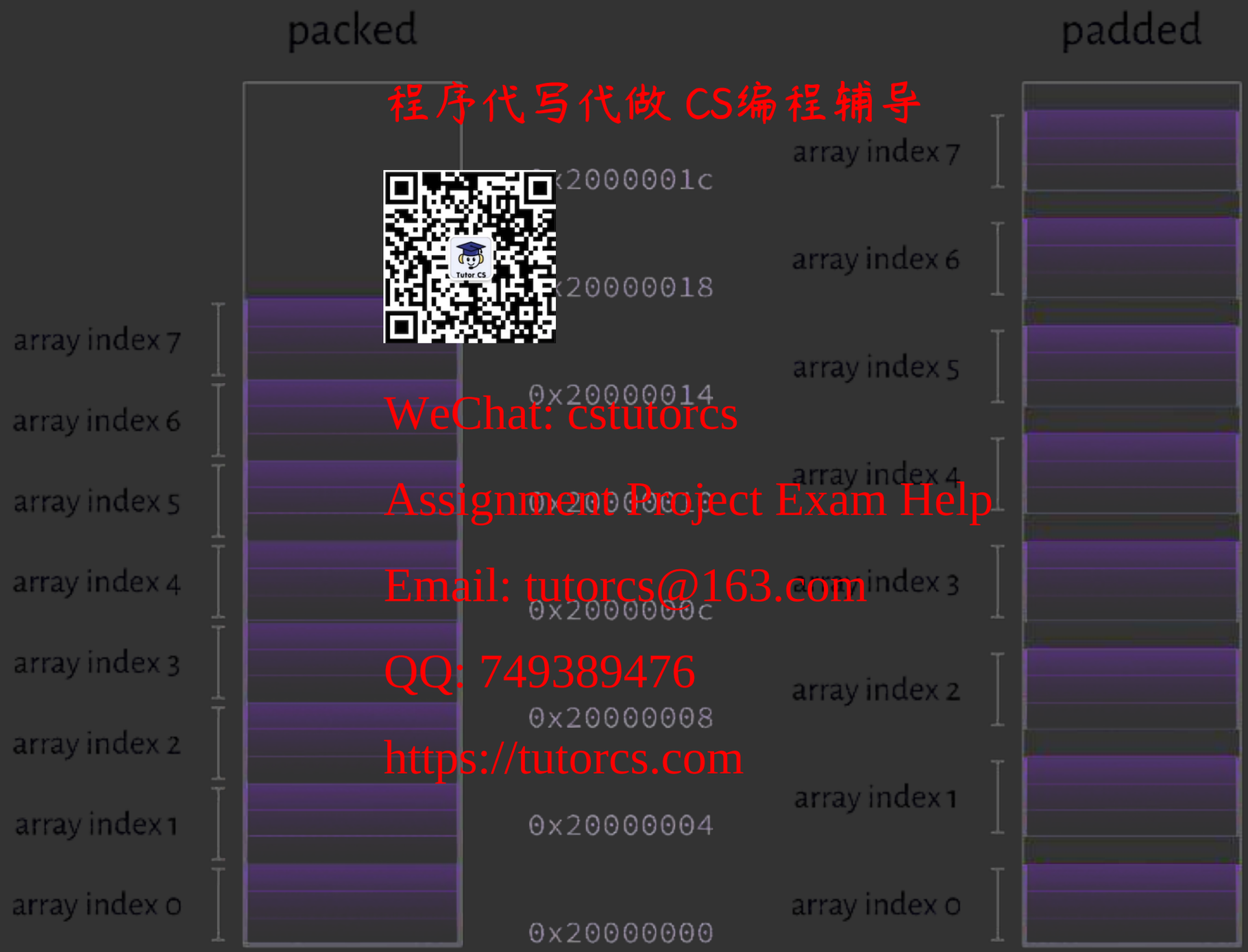
WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com



程序代写代做 CS编程辅导
WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

程序代写代做 CS编程辅导

abstract data type

WeChat: cstutorcs

Assignment Project Exam Help

VS

Email: tutorcs@163.com

QQ: 749389476

data structure

https://tutorcs.com

Working with arrays

程序代写代做 CS编程辅导

they don’t fit in registers, so we h[illegible]rate on them one element at a time

we need **loops**!

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Array sum example

程序代写代做 CS编程辅导

given an array of (word-size) inte[illegible]he sum of the elements from `from_index` to `to_index`

```
int sum(int array[], int from_index, int to_index){
  int acc = 0; // the "accumulator" variable
  for(int i = from_index; i <= to_index; i++){
    acc += array[i];
  }
  return acc;
}
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

The array in memory

程序代写代做 CS编程辅导

```
  .data
array:
  .word 2, 3, 0, 0, 6, 0
```

back-of-the-envelope maths: the sum is 14 (0xE)

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Array sum setup

程序代写代做 CS编程辅导

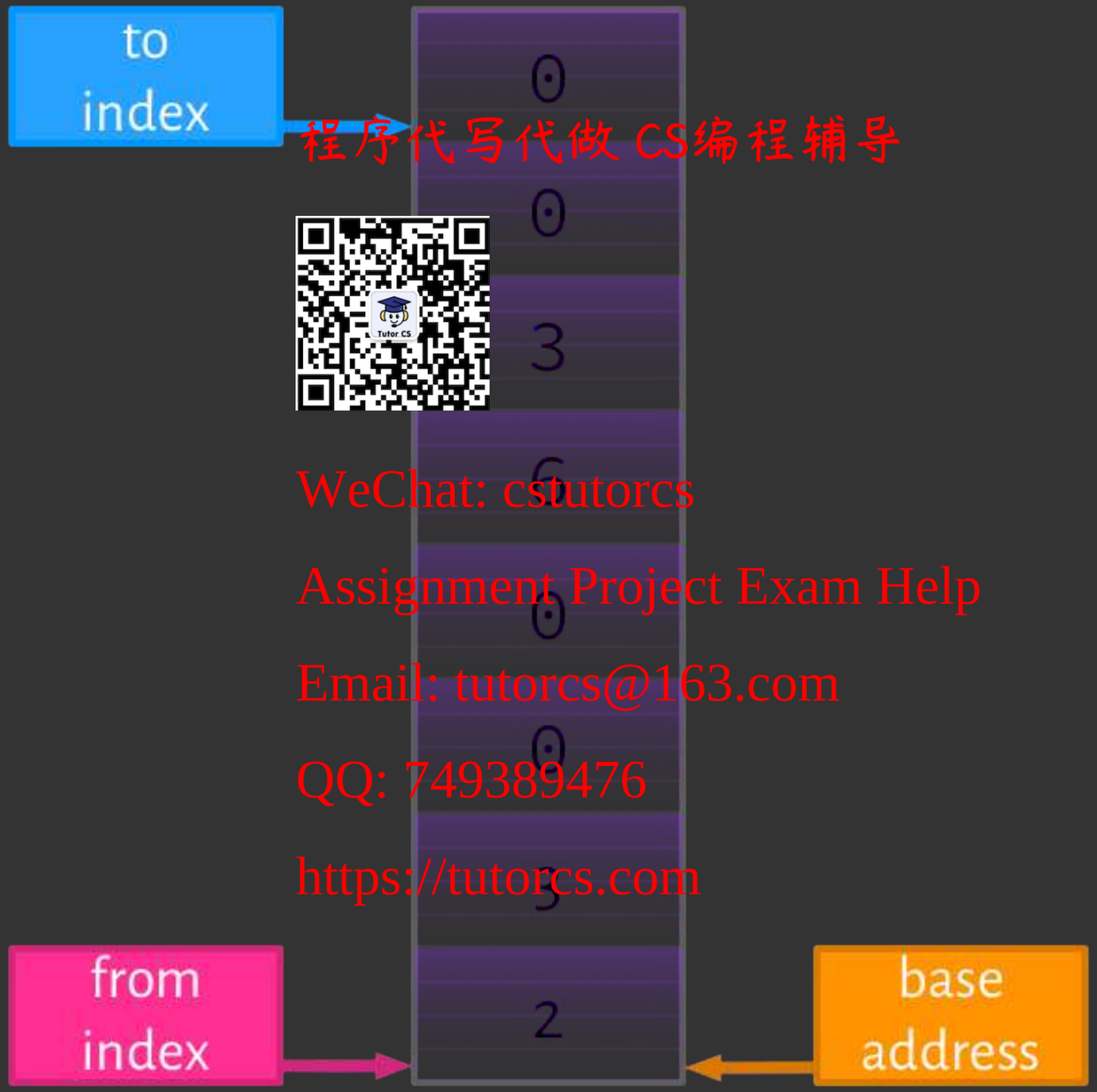
For the following examples, assu

- the base address of the array i
- the from (start) index is in `r1` (0 in the setup code below)
- the to (end) index is in `r2` (7 in the setup code below)

WeChat: cstutorcs
Assignment Project Exam Help

```
ldr r0, =array @ base address
mov r1, 0      @ from_index
mov r2, 7      @ to_index
```

Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com



程序代写代做 CS编程辅导

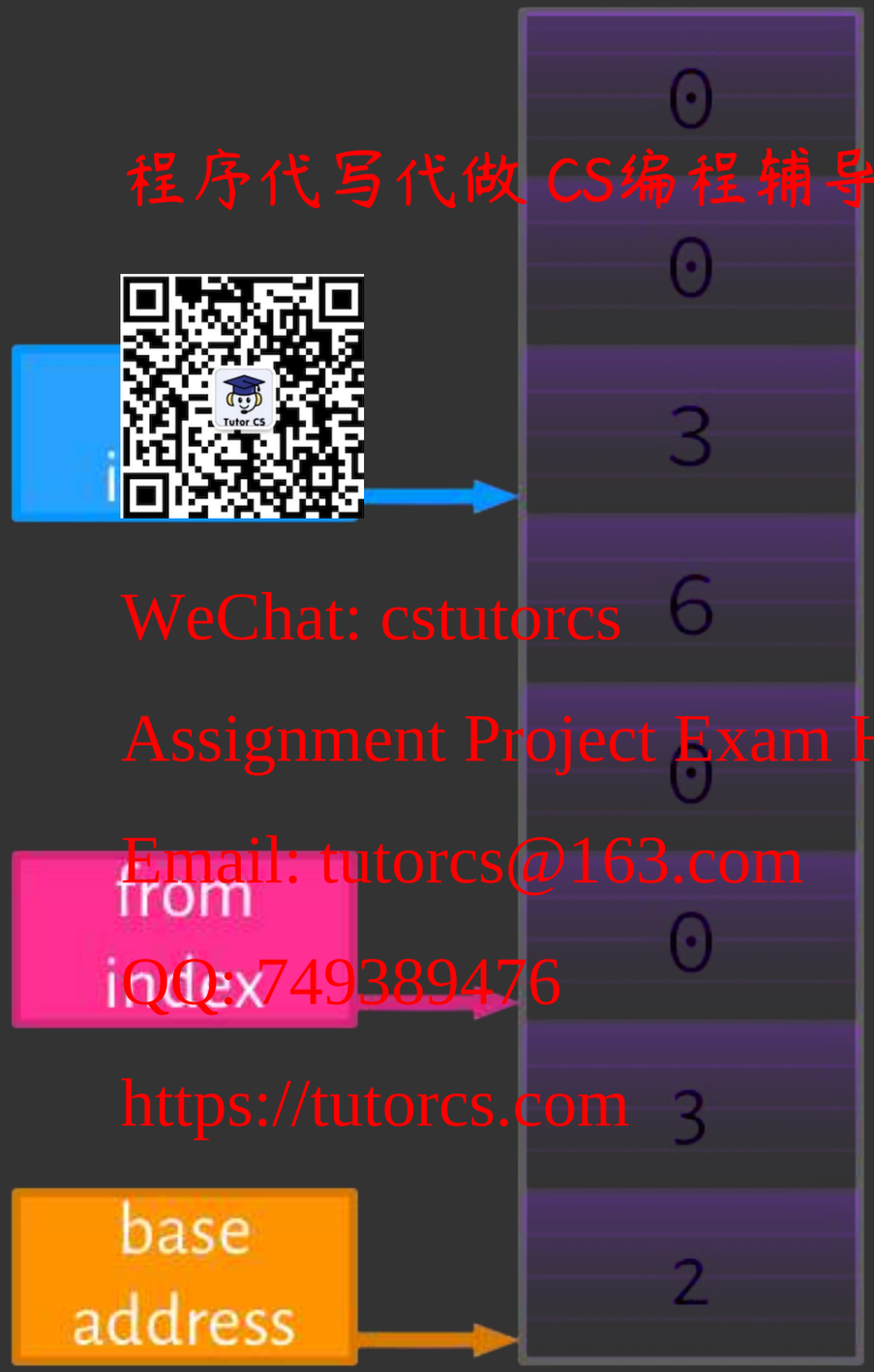
WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com



程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Array sum #1

程序代写代做 CS编程辅导

```
@ setup
  mov r3, 0 @ "accumulator"
  mov r4, 4 @ element size

array_sum:
  mul r5, r1, r4    @ calculate offset
  ldr r6, [r0, r5]  @ load from offset
  add r3, r6        @ update accumulator
  add r1, 1         @ increment index
  cmp r1, r2        @ keep looping?
  ble array_sum

@ cleanup
  mov r0, r3
```

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

2 instructions in setup, 6 in loop

Array sum #2

程序代写代做 CS编程辅导

```
@ setup
  mov r3, 0 @ acc

array_sum:

  @ load with shifted index register
  ldr r6, [r0, r1, lsl 2]
  add r3, r6      @ update running total
  add r1, 1       @ increment index
  cmp r1, r2      @ keep looping?
  ble array_sum

@ cleanup
end_array_sum:
mov r0, r3
```

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

1 instruction in setup, 5 in loop, no need to explicitly calculate the offset (but size must be power of 2)

Array sum #3

程序代写代做 CS编程辅导

```
@ setup
  mov r3, 0      @ acc
  lsl r1, r1, 2 @ change inc set
  lsl r2, r2, 2 @ change inc set

array_sum:
  ldr r6, [r0, r1]
  add r3, r6     @ update running total
  add r1, 4      @ increment index
  cmp r1, r2
  ble array_sum

@ cleanup
mov r0, r3
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

3 instruction in setup, 5 in loop

uses byte offset instead of element index

talk

程序代写代做 CS编程辅导

any more ideas for how we coulc s up?

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Array sum #4

程序代写代做 CS编程辅导

```
@ setup
  mov r3, 0       @ acc
  lsl r1, r1, 2   @ change in[illegible]fset
  add r1, r0, r1  @ address o[illegible]ement

  lsl r2, r2, 2   @ change index -> offset
  add r2, r0, r2  @ address of to element

array_sum:
  ldr r6, [r1], 4 @ load & post-index r1
  add r3, r6      @ update running total
  cmp r1, r2
  ble array_sum

@ cleanup
  mov r0, r3
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

5 instructions in setup, **only 4 in loop**, note the **load with post-index**

On optimisation…

程序代写代做 CS编程辅导

loops are often the “hot” part of , therefore worth optimising

optimising compilers will do som ings to get the most optimised code

in general, write simple code first, and optimise later (if necessary)

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Gotchas

程序代写代做 CS编程辅导

- don't forget about **endianness**
- remember, `ldr`/`str` still ju **rds**, not elements (it's convenient if your elements are word-sized, but f they're not)
- no bounds checking so far!
- we weren't careful about the **AAPCS** in our array sum examples earlier

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Memory allocation for arrays

程序代写代做 CS编程辅导

static: memory is set aside at co (e.g. in the `.data` section)

- *pro*: allocation is already done program starts
- *con*: need to know the size of the array in advance

WeChat: cstutorcs

dynamic: memory is made available (e.g. on the stack) at run-time

Assignment Project Exam Help

- *pro*: can pick the best size at runtime, can re-size

Email: tutorcs@163.com

- *con*: takes time (while program is running)

QQ: 749389476

we'll return to this later…

https://tutorcs.com

Better(?) array data structures

程序代写代做 CS编程辅导

The arrays we've looked at so far are bare-bones; containing just the raw data (at runtime, anyway)

there are several improvements

- null-termination
- store array size alongside the data
- can we make a resizeable array?

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

程序代写代做 CS编程辅导

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Where are the array(s)?

程序代写代做 CS编程辅导
WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Records

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Records are for collections of heterogeneous data

程序代写代做 CS编程辅导

Wikipedia says:

A **record** is a collection of **fields**, possibly of different data types, typically in fixed number and sequence. The fields of a record may also be called members, … or elements, though these risk confusion with the elements of a collection.

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Overloaded names!

records might also be referred t s, tuples, objects

(although those terms can also mean different things)

context matters

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Array

程序代写代做 CS编程辅导
WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Simple examples

程序代写代做 CS编程辅导

Suppose we are making charact a computer game:

- HP (word)
- mana (word)
- name (16-byte, null-terminated ASCII array)

WeChat: cstutorcs

Or a basic synthesizer

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

- frequency (word)
- phase (word)
- amplitude (word)
- type (halfword)

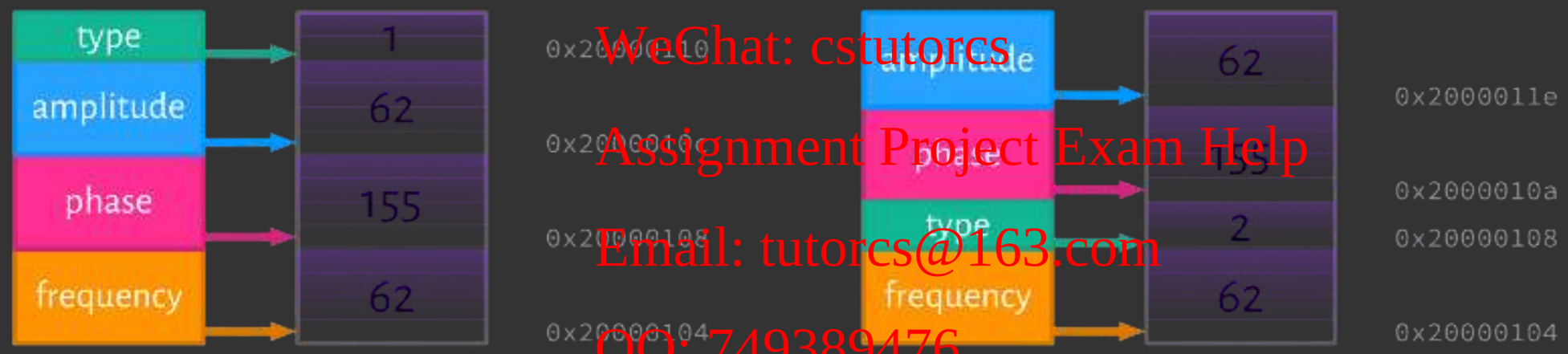
程序代写代做 CS编程辅导

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Field ordering

what's the difference here?

程序代写代做 CS编程辅导



WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

the address of the first element is all you need

程序代写代做 CS编程辅导

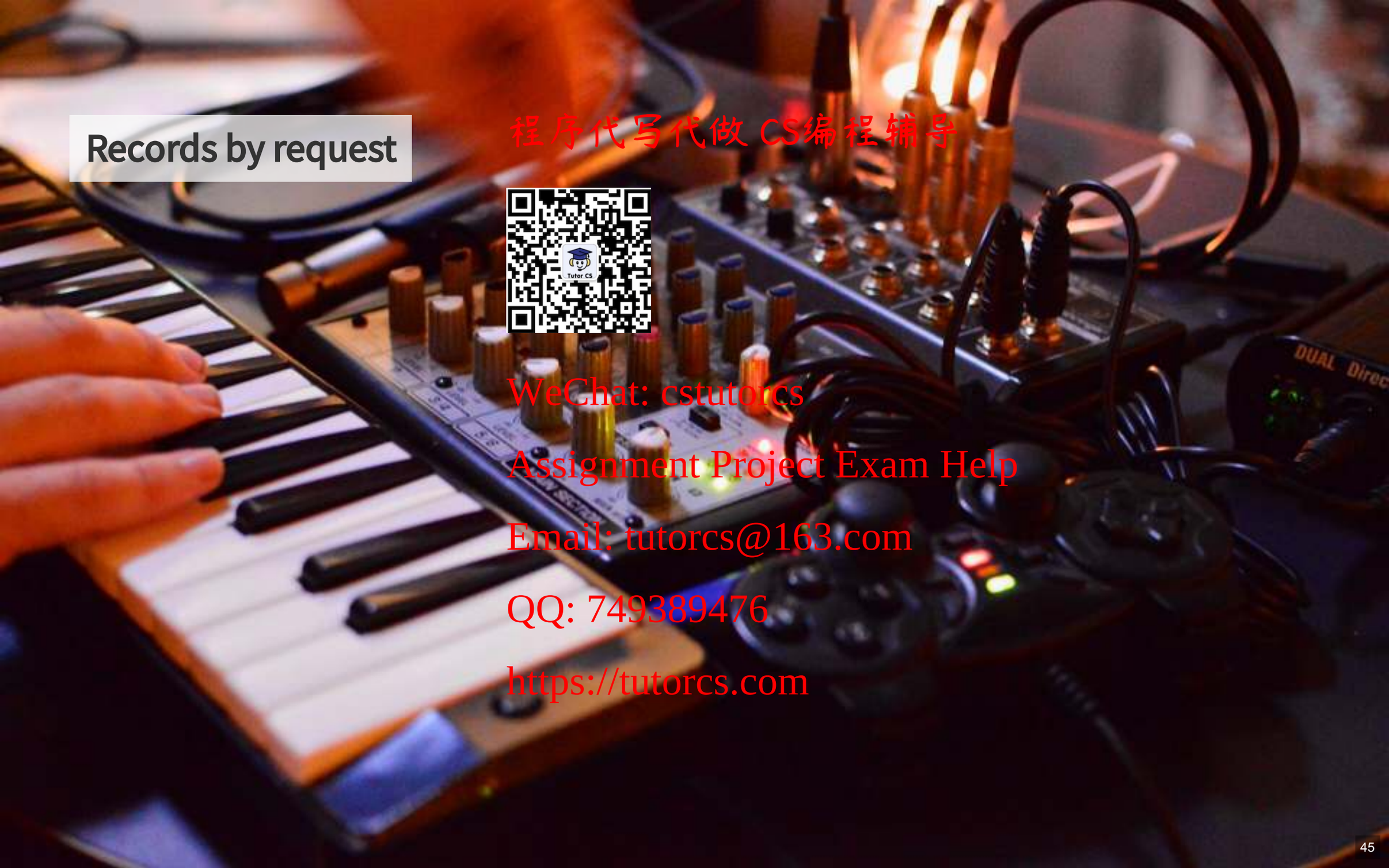
WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com



Records by request

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Composite data structures

程序代写代做 CS编程辅导

Things really get interesting and [illegible]n we combine things

- records inside records (nested
- records of arrays
- arrays of records

WeChat: cstutorcs

Assignment Project Exam Help

Are these things OO *objects*? No. They have the variables but not the methods.

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Memory Allocation

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

talk

程序代写代做 CS编程辅导

what does it mean to allocate memory for an array?

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

`smalloc` (silly memory allocate)

Let's make a silly memory alloc

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Describing the `smalloc` function

程序代写代做 CS编程辅导

At a *minimum*, our memory allocation must

- take (as an input parameter) the number of bytes to allocate
- return a memory address which points to that many bytes of **free** memory

WeChat: cstutorcs

there are many other things it could do, but this will do for now

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Silly Malloc

程序代写代做 CS编程辅导

```
@ r0 param is number o                    to allocate returns a memory addre
.type smalloc, %functi
smalloc:

  push {r0} @ just save r0 for a sec
  ldr r3, =smalloc_value
  ldr r0, [r3] @ get latest free memory address, this is the ret
  pop {r1} @ get back parameter
  add r1, r1, r0 @ new "smalloc value", just move it along by re
  str r1, [r3] @ save new smalloc_value
  bx lr
.size smalloc, . - smalloc

.data
smalloc_value:
.word 0x20000004 @ 4 because it starts AFTER this word!
```

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Real `malloc`

程序代写代做 CS编程辅导

real operating systems provide a function for dynamically allocating memory which sortof works like this, but tter:

- allows you to “release” memory after you’re done with it
- keeps track of the memory allocations using metadata
- (tries to) gracefully fail when it can’t give you enough memory

although it’s implementation dependent

WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Data structures and argument passing

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

talk

程序代写代做 CS编程辅导

how would you pass an array or an argument to a function? how about as a return value?

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

it's alive

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Records as function parameters

程序代写代做 CS编程辅导

imagine an `is_alive` function

- takes one parameter: a *hearth* *aracter* data structure
- returns a (word-sized) `0` if the character is dead, and `1` if the character is alive

WeChat: cstutorcs

```
  .data
zoltan_the_magnificent:
@ initialise hp and mana
  .word 100, 200 @ more mana because wizard
```

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

It's alive #1: pass character in registers

程序代写代做 CS编程辅导

```
is_alive:
  movs r0, r0 @ cheeky trick
  mov r0, 0
  ble end_is_alive

  mov r0, 1
end_is_alive:
  bx lr

@ call
  ldr r2, =zoltan_the_magnificent
  ldr r0, [r2]     @ hp
  ldr r1, [r2, 4] @ mana
  bl is_alive
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

It's alive #2: pass character on stack

程序代写代做 CS编程辅导

```
is_alive:
  pops {r0,r1} @ read args c
  mov r0, 0
  ble end_is_alive

  mov r0, 1
end_is_alive:
  bx lr

@ call (including copy record onto the stack)
@ for larger records this might require a loop
  ldr r2, =zoltan_the_magnificent
  ldr r0, [r2]
  ldr r1, [r2, 4]
  push {r0,r1}
  bl is_alive
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

It's alive #3: pass character by reference

程序代写代做 CS编程辅导

```
is_alive:
@ load hp using the address
@ no need to load mana
  ldr r1, [r0]

  movs r1, r1
  mov r0, 0
  ble end_is_alive
  mov r0, 1
end_is_alive:
  bx lr

@ call (pass only address of "zoltan" record)
  ldr r0, =zardok
  bl is_alive
  cmp r0, 1
  beq whee
  b no
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Gotchas

程序代写代做 CS编程辅导

the records in these examples are small enough that they *could* be passed in registers, but large ones can't be

be aware of **stack discipline**

WeChat: cstutorcs

pass by reference: no copying, but the caller can mess with the "source" data

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Further reading

Inside memory management: T, tradeoffs, and implementations of dynamic allocation

A look at how malloc works on mac

Plantz: Chapter 15 - Data Structures

程序代写代做 CS编程辅导
WeChat: cstutorcs
Assignment Project Exam Help
Email: tutorcs@163.com
QQ: 749389476
https://tutorcs.com

Questions?

程序代写代做 CS编程辅导

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com