270069 - PCA - Architecture-Aware Programming

Coordinating unit: 270 - FIB - Barcelona School of Informatics
Teaching unit: 701 - AC - Department of Computer Architecture
Academic year: 2018
Degree: BACHELOR’S DEGREE IN INFORMATICS ENGINEERING (Syllabus 2010). (Teaching unit Optional)
ECTS credits: 6  
Teaching languages: Catalan, Spanish

Teaching staff

Coordinator: - Daniel Jimenez Gonzalez (djimenez@ac.upc.edu)

Prior skills

The requirements provided for.

Requirements

- Prerequisite AC

Degree competences to which the subject contributes

Specific:
CEC2.2. To program taking into account the hardware architecture, using assembly language as well as high-level programming languages.
CEC2.3. To develop and analyse software for systems based on microprocessors and its interfaces with users and other devices.
CEC3.1. To analyse, evaluate and select the most adequate hardware and software platform to support embedded and real-time applications.
CEC3.2. To develop specific processors and embedded systems; to develop and optimize the software of these systems.

General:
G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.
Learning objectives of the subject

1. Follow a methodology of optimization to reduce the execution time of an application in a specific processor; first identifying the parts more expensive, then evaluating whether or not (Amdahl's law) its optimization is worth, and finally perform the optimization if the performance gain is reasonable.

2. Identify and recognize the optimizations the compiler can do, either by the basic knowledge that the student has of the gcc compiler, the analysis of assembler code generated, or because the student does profiling or instrumentation.

3. Choose the tools and their options most suitable to generate the binary code, analyze the binary code generated and the execution of that code (by profiling, instrumentation or tracing); in addition to choosing metrics for this analysis and the environment more suitable for evaluation.

4. Define latency, repetition rate and throughput of an instruction; and apply this knowledge in optimizing of the codes to choose the most appropriate operation in each case.

5. Apply, in the proper way, optimization techniques to reduce and / or avoid long latency operations, reducing the execution time of the application.

6. Reduce the time allocated for breaks in code, using techniques that reduce the number of jumps.

7. Characterize the memory hierarchy of the computer with either a manual or studying it empirically with codes and tools for profiling and tracing.

8. Apply, in the proper way, optimization techniques to make more efficient use of processor memory hierarchy, reducing the execution time of the application.

9. Identify if a code is vectorizable and vectorized it if it means reducing the execution time of the application.
11. Analyze and identify if a code can be accelerated using custom hardware and use the high level tools provided in the course to define the specific hardware from C language with very simple pragmas.
12. Analyze and optimize a medium sized application along the semester, applying the optimization techniques and methodology followed in the course, in coordination with the rest of their team members.
13. Lead a group of students to obtain the best solution to an optimization problem or exercise of analysis given, considering the solutions of the rest of his team members and after a discussion among all team members.

| Study load |
| --- | --- | --- |
| **Total learning time:** 150h | Theory classes: 15h 10.00% |
| | Practical classes: 15h 10.00% |
| | Laboratory classes: 30h 20.00% |
| | Guided activities: 6h 4.00% |
| | Self study: 84h 56.00% |
### Introduction: Architecture-Aware Programming

Degree competences to which the content contributes:

Description:
> What we mean by Architecture-Aware Programming and in what cases it is worth it. Optimization strategies. Performance evaluation.

### Programming and Optimization Tools

Degree competences to which the content contributes:

Description:
> Performance measurement programs. Compilation options. Study of dynamic behavior of programs. Use of high level tools to accelerate (optimize) some part of the code in hardware.

### Long latency operations

Degree competences to which the content contributes:

Description:
> Alternatives to run some high cost operations and/or long latency. Replacement of complex operations for simple operations. Memoization of results. Detection of trivial cases. Use of operating system resources.

### Control flow optimization

Degree competences to which the content contributes:

Description:
> Detection and elimination of critical breaks. Inlining. Loop Unrolling, Collapsing, Fusion, etc.

### Memory Conscious Optimizations

Degree competences to which the content contributes:

Description:
> Size and alignment of data. Division of data into blocks of size proportional to the cache (blocking). Memory disambiguation. Definition of data structures and pragmas to improve the performance of the application both in a specific hardware or a general purpose processor.

### Vector Extensions and use of custom hardware to "vectorize"

Degree competences to which the content contributes:
### Description:

SIMD extensions. Identification of codes that can be vectorized. Inserting source code vector in C. Use of tools of High Level Synthesis to accelerate (optimize) in hardware the most time consuming parts of the program exploiting the same idea of vectorization.
## Planning of activities

<table>
<thead>
<tr>
<th>Planning of activities</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Programming conscious</td>
<td>4h</td>
</tr>
<tr>
<td>architecture</td>
<td></td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
<tr>
<td>Actively participate in a session-class</td>
<td></td>
</tr>
<tr>
<td>exhibition participatory 2 hours of theory</td>
<td></td>
</tr>
<tr>
<td>and problems. (2 hours). Studying at home</td>
<td></td>
</tr>
<tr>
<td>and do the exercises assigned topic to be</td>
<td></td>
</tr>
<tr>
<td>proposed as an individual homework.</td>
<td></td>
</tr>
<tr>
<td><strong>Specific objectives:</strong></td>
<td>1</td>
</tr>
<tr>
<td><strong>Hours</strong></td>
<td>20h</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
<tr>
<td>Actively participate in a session-class</td>
<td></td>
</tr>
<tr>
<td>exhibition participatory 2 hours of theory</td>
<td></td>
</tr>
<tr>
<td>and problems. (2 hours). Studying at home</td>
<td></td>
</tr>
<tr>
<td>and do the exercises assigned topic to be</td>
<td></td>
</tr>
<tr>
<td>proposed as an independent work.</td>
<td></td>
</tr>
<tr>
<td><strong>Specific objectives:</strong></td>
<td>2, 3, 11</td>
</tr>
<tr>
<td><strong>Hours</strong></td>
<td>17h</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
<tr>
<td>Participar activament en una sessió de</td>
<td></td>
</tr>
<tr>
<td>classe expositiva-participativa de 2 hores</td>
<td></td>
</tr>
<tr>
<td>de teoria i problemes. (2 hores). Estudiar</td>
<td></td>
</tr>
<tr>
<td>a casa el tema assignat i realitzar els</td>
<td></td>
</tr>
<tr>
<td>exercicis que s'hagin proposat com a treball</td>
<td></td>
</tr>
<tr>
<td>autònom.</td>
<td></td>
</tr>
<tr>
<td><strong>Specific objectives:</strong></td>
<td>1, 2, 4,</td>
</tr>
<tr>
<td>5, 13</td>
<td></td>
</tr>
<tr>
<td><strong>Hours</strong></td>
<td>29h</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Hours</strong></td>
<td>29h</td>
</tr>
</tbody>
</table>

### Theory classes: 1h

- Practical classes: 0h
- Laboratory classes: 2h
- Guided activities: 0h
- Self study: 1h

### Theory classes: 2h

- Practical classes: 2h
- Laboratory classes: 6h
- Guided activities: 0h
- Self study: 10h

### Theory classes: 2h

- Practical classes: 2h
- Laboratory classes: 4h
- Guided activities: 0h
- Self study: 9h

### Theory classes: 3h

- Practical classes: 2h
- Laboratory classes: 6h
- Guided activities: 0h
- Self study: 18h
### Description:
Actively participate in a session-class exhibition participatory 2 hours of theory and problems. (2 hours). Studying at home and do the exercises assigned topic to be proposed as an independent work.

**Specific objectives:**
1, 2, 3, 6, 11, 13

### Control 1-T

| Hours | Guided activities: 1h  
                | Self study: 2h |
|-------|---------------------|

#### Description:
Issues 1-3.

#### Specific objectives:
1, 2, 3, 4, 5, 7, 11

### Control-1-P

| Hours | Guided activities: 1h  
                | Self study: 2h |
|-------|---------------------|

#### Description:
Lessons 1-3

#### Specific objectives:
1, 2, 3, 4, 5, 7, 11

### Optimizations conscious memory

| Hours | Theory classes: 2h  
                | Practical classes: 3h  
                | Laboratory classes: 6h  
                | Guided activities: 0h  
                | Self study: 14h |
|-------|---------------------|

#### Description:
Actively participate in a session-class exhibition participatory 2 hours of theory and problems. (2 hours). Studying at home and do the exercises assigned topic to be proposed as an independent work.

#### Specific objectives:
1, 2, 3, 7, 8, 11, 13
Vector Extensions and custom hardware usage

**Description:**
Actively participate in a session-class exhibition participatory 2 hours of theory and problems. (2 hours). Studying at home and do the exercises assigned topic to be proposed as an independent work.

**Specific objectives:**
1, 2, 3, 9, 11, 13

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 3h</th>
<th>Practical classes: 4h</th>
<th>Laboratory classes: 6h</th>
<th>Guided activities: 0h</th>
<th>Self study: 22h</th>
</tr>
</thead>
</table>

Control 2-T

**Description:**
All subjects. More centered on 5-6, but any previous concept may be evaluated.

**Specific objectives:**
1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13

<table>
<thead>
<tr>
<th>Hours</th>
<th>Guided activities: 1h</th>
<th>Self study: 2h</th>
</tr>
</thead>
</table>

Control-2-P

**Description:**
All topics. More focused on topics 4-6 but any previous concept can come out.

**Specific objectives:**
1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13

<table>
<thead>
<tr>
<th>Hours</th>
<th>Guided activities: 1h</th>
<th>Self study: 2h</th>
</tr>
</thead>
</table>

Resolution of doubts related to the practices or visit a research center.

**Specific objectives:**
1, 2, 3, 5, 6, 8, 9, 11, 12

<table>
<thead>
<tr>
<th>Hours</th>
<th>Theory classes: 0h</th>
<th>Practical classes: 0h</th>
<th>Laboratory classes: 0h</th>
<th>Guided activities: 3h</th>
<th>Self study: 2h</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Final Exam</strong></td>
<td><strong>Hours:</strong> 3h</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td>Guided activities: 3h</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Exam. Outside of class time.</td>
<td>Self study: 0h</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Specific objectives:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Qualification system

In the evaluation of the student involved five components:

(1) Laboratory exercises: L - 40% of the subject mark.

There is a lab practice per lesson and there may be more than one session per lab practice. Each practice will be 20% of the overall lab mark.

\[ L = 0.2 \cdot L_2 + 0.2 \cdot L_3 + 0.2 \cdot L_4 + 0.2 \cdot L_5 + 0.2 \cdot L_6 \]

The solution of lab exercises will be a brief report (between 1 and 5 pages), source codes and scrips that must be delivered to the teacher via the Raco to be evaluated. Also, some of the codes may be asked to be delivered to a server where they will be compiled and checked if they correctly accelerate the application.

(2) Exercises class of problems: P - 10% of the subject mark.

These exercises are requested by the teacher in class of problems and must work in teams and presented to Athena Forum or via Raco.

(3) Partial control mark: C - 50% of the subject mark.

During the course students must answer two controls: C1, C2. Its objective is to assess whether the student has achieved assimilate code optimizations explained so far.

\[ C = 0.35 \cdot C_1 + 0.65 \cdot C_2 \]

(4) Final Exam Mark: F

Final: Final examination will include theoretical and practical cases. This exam is only compulsory for those students that have not pass the controls.

(5) Extra Challenge Point: cha

Challenge: We propose an optimization exercise delivered via a web portal or via Raco that will be awarded function of the final ranking position. The student with the best solution's final grade will be increased by 1 point (faster solution), 0.9 points (second quickest solution),... and 0.1 points (tenth quickest solution) respectively. These extra points will be only applied to those student that has C>=5.0.

The course can be approved in two ways:

The grade exam (NEX) can be achieved in two ways, either by continuous assessment or final examination:

If \((C > 5)\)

\[ NEX = \text{MAX} (C, F) \]

otherwise:

\[ NEX = \text{MAX}(0.25 \cdot C + 0.75 \cdot F, F) \]

The final grade is:
Final Mark = MIN (0.50 * NEX + 0.10*P + 0.40*L + cha 10)

The competition evaluates each student considering cross their teamwork, leadership in finding effective solutions, alternative, creative and innovative to the problems to be solved which the student is responsible. That is, it will be evaluated their ability to find a solution to be a successful combination of the solutions found by a problem or the solution. Also part of this assessment to participate in finding creative solutions to problems that can not be responsible.

**Bibliography**

**Basic:**