MEMORY WASTAGE ANALYSIS USING VARIABLE ALIGNMENT

Vaishali, R.[1], Shathana, R.[1], Lojenaa, N.[1], Kartheeswaran, T.[2]

[1] *Department of Physical Science, Faculty of Applied Science, Vavuniya Campus of the University of Jaffna, Sri Lanka*
[2]*Department of Computer Science and Technology, Faculty of Science and Technology, Uva Wellasa University of Sri Lanka*

Abstract

In present world, people are proficient with interactive devices. They prefer to use simple faster devices. The responsibility falls on the hands of developers. So they must be prudent while programming a device. Mainly they should consider about memory and execution time. When more memory is wasted, then it will make an impact on speed and further becomes inefficient for allocating memory for other purposes of the processes or programs that runs on the same host. Hence developers must be aware of wastage and where it occurs. It's a very simple product that help to student to find the memory efficiency of their laboratory practical (programming) and also assists the developers to analyze the wastage of memory that is caused while programming. Then after, they can manually rearrange the code for the purpose of reducing the wastage of memory. The project was designed to prepare the students in efficient programming in laboratory practical. The extreme goal of the project is to provide the knowledge on memory management to Information Technology (IT) undergraduates from the learning state.

Keywords: memory, wastage, analyze, programming

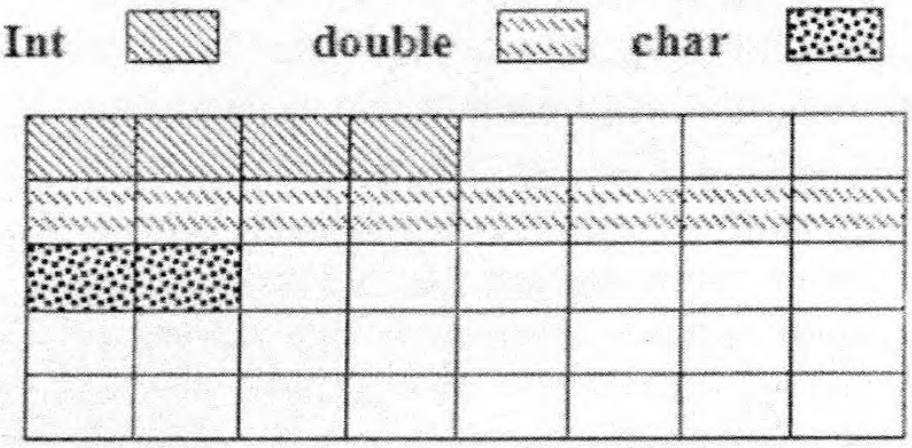
1. INTRODUCTION

In present world, people are proficient with interactive devices. They prefer to use simple faster devices. The responsibility falls on the hands of developers. So they must be prudent while programming a device. Mainly they should consider about wastage of memory and time. When more memory is wasted for mobile or embedded device programming, then it will make an impact on speed and further becomes inefficient. Hence developers must be aware of wastage and where it occurs. Generally when a variable is declared, a portion of memory will be allocated for it. Wastage of memory may occur when it's declared while programming in an unorganized manner. This will be an issue while writing programs for embedded or mobile devices such as mobile phones, digital camera or any other hand held smart devices. As mobile devices have limited amount of memory, writing program for mobile processors not be unnecessarily large and not be consume more memory. More memory consuming programs may cause difficulty in memory allocation for other features and cause inefficient programming by becoming slower processing. Hence, we should concentrate memory efficient programming due to the lack of memory in small devices unlike in desktop or server computers. This simple software tool will somehow assist to the programming students to learn memory efficient programming and track their memory usage.

Furthermore the tool helps to programmers also to identify the memory wastage by the in-ordered variable declarations. So, they can align variables in order to reduce the memory wastage. Prepare the students to write memory efficient programs from the learning state will make more efficient developer and Software engineers in future

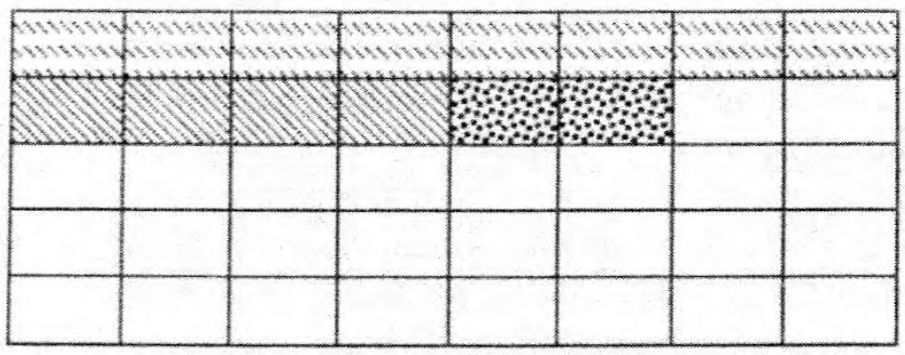
2. METHODOLOGY

We designed an efficient software tool using Visual Studio 2013 with C# which helps to find the memory wastage by analyzing the memory alignment of the given program. For an example if the memory blocks are 8 bytes each, then the memory allocation of the below given program (Figure 1. Program 1) can be calculated using Figure2 (Figure 2. Memory layout).




```
public class HelloWorld
{
    int no;
    double amount;
    charopt,opr;
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

The Figure 2 clearly shows that there is a 4 Bytes memory has been wasted by improper alignment of variables in the program1. The wastage can be reducing as 0 Bytes by realigning the variables. The program2 has been written with reduced memory wastage using variable alignment in an efficient manner.



```
public class HelloWorld
{
    double amount;
    int no;
    charopt,opr;
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

The software tool will analyze the program code and find the memory wastage by tracking the order of variables. Using the software tool students and also programmers analyze their codes and find the memory wastage of their programs. This memory space may allocate in ***.bss*** segment or in the ***stack*** segment of the memory. The programmers can concentrate on variable alignment and reduce their memory wastage by realigning the variables. Currently the technological developments move towards portable and mobile devices. The software tool will be useful for mobile and android programmers to write the memory efficient programs. The software tool has been validated using different kind of program codes for testing.

2.1 Implementation

This software tool was basically implemented by the following procedure

- The total number of lines in the source code is counted.
- Each line is tested whether it contains any data types. And if so, the number of variables in the particular data type is counted and printed.
- Meantime, it was virtually aligned according to the selected bit-alignment and memory wastage is calculated.
- At the end of counting, memory wastage according to bit alignment will be displayed.

3. RESULTS AND DISCUSSION

There are several studies has been done in this domain in more deeper level but for the undergraduate level it's difficult to implement and used the tools according to their studies. Most of the studies are focused on dynamic memory allocations, using dynamic memory allocation functions such as malloc(), alloc(), realloc(), etc... But the optimization using variable alignment is not considered [1]. Some methods are driven by tracking the memory usage changes in programming [2]. The studies are based on some applications which run in embedded systems such as speech recognition etc.[2].The concept that used here to make the students aware while learning programing regarding memory efficient programs in CPU and Embedded system perspective and train the students in efficient programing for almost all the leading programming languages used in university level programming practical.

The developed software tool will show you the amount of memory wastage according to the selected block size of memory. The software tool shows the alignment of variables from the given code with the recoverable wasted memory. So the tool helps to the programmers to easily identify and rearrange the variables. They can check the memory wastage often and ensure the reduced memory wastage. Therefore, an efficient programming code can be developed for mobile and embedded devices.

After memory wastage is calculated, manually the developer can simply realign the program based on reducing the wastage. The memory alignment of a program not only affects the memory consumption of a processor but also execution time of the program. Execution time of a program can be simply calculated by the default time calculation method. The deviation in memory allocation and execution time can be observed before and after the re-arrangement of variables. The impact of memory allocation is easy to identify in a simple program; but it shows its effect on large programs. The students and developers may not be aware of random declaration of variables while developing their programs. They'll be only aimed at development and completion of code. As the program is large, there are lots of variables may be declared and scattered all over the program code. This will show the impact while executing the program. Hence, this tool will help them to identify how their reluctance in declaration of variables has caused memory wastage and slow in execution. This tool will show line by line in which, what type of variable and the number of variables present. And also shows the memory wastage according to the preferred bit alignment. So they can easily identify the variables where it is in the program and may remove unwanted variables.

```
long startTime = System.nanoTime();
methodToTime();
long endTime = System.nanoTime();

long duration = endTime - startTime;
```

Figure 4: program

Code	Language	Memory wastage
Insertion sort	Java	4
Insertion sort	C#	0
Factorial	C	0
Bresenham circle	Java	0
Insertion sort	C#	0
Reflection of Shape	Java	13

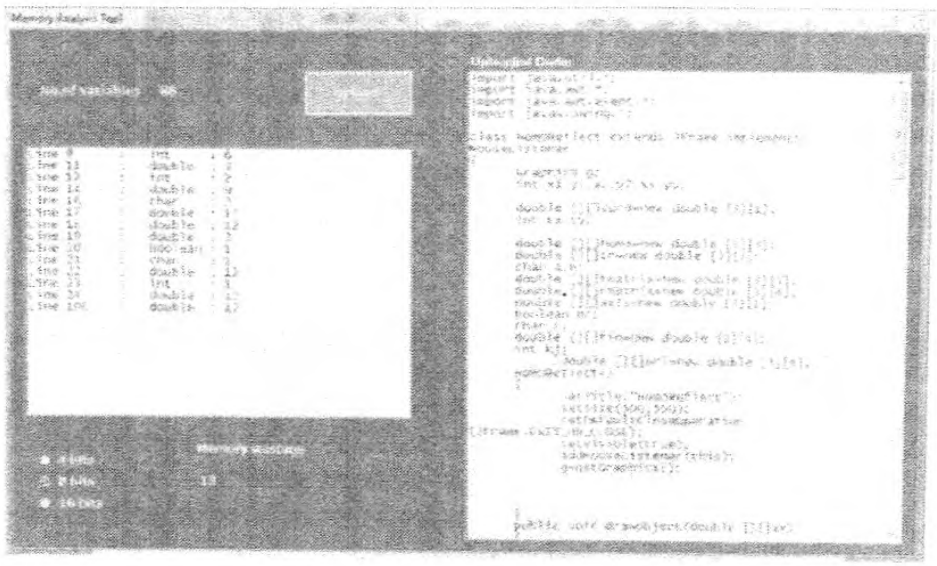


Figure 05: Interface

4. CONCLUSION

The main purpose of the designed tool is to observe the efficiency of the programs which are developed by the students during practical lectures. The students can check their programs for memory wastage and they can realign their variables to reduce the waste of memory and unnecessary execution delay. These practices will lead the students to learn efficient programming from the beginning. This product may help to overcome one of the existing problems in software development for embedded or mobile devices when they are unnecessarily unknowingly allocating memory for variables. The deliverable tool will help to user to reduce the time consumption of manual activities like finding memory wastage. And also it helps to reduce the wastage of memory. As a result of this automation, manual workload is reduced and the student's efficiency in programming can be increased. The study is currently ongoing to suggest the students and programmers an efficient realignment method for their codes using this tool and further the tool will be improved using more programming languages ,compiler level calculations and pre and post alignment execution time calculation using the above given Figure5 (Figure 5. Snippet for execution time calculation). The execution time improvements make the end user to be aware about the effects of memory wastage in execution time of a program.

REFERENCES

S. Yang and i. Verbauwhede "methodology for memory analysis and optimization in embedded systems", proc. Gspx embedded signal processing conference, 2004

C, srilatha, and guru rao c. V. "a novel approach for estimation and optimization of memory in low power embedded systems". *Ijcte* (2009): 581-587. Web.

intel. (n.d.). Retrieved 04 23, 2015, from software intel: https://software.intel.com/en-us/articles/coding-for-performance-data-alignment-and-structures

Microsoft. (n.d.). Retrieved 04 27, 2015, from msdn: https://msdn.microsoft.com/en-us/library/dd383503(v=vs.110).aspx