Procedural modelling of buildings
Michal Rosa | Supervisor: Mgr. Jan Beneš
Charles University in Prague, Faculty of Mathematics and Physics
Department of Software and Computer Science Education

Abstract
In this bachelor thesis we focus on implementation of a standalone application for procedural modelling of buildings. We then test our implementation by modelling several buildings based on buildings in real life. Apart from our implementation, we provide a short overview of already existing methods. The thesis also includes programming and user documentation.

Introduction
In our work we use computer graphics to model simple or more complex building structures. For example, when creating a movie or a game we need to model many buildings or whole cities. Manual modelling on such scale would be inefficient, time consuming and it would require a considerable amount of human resources, time, computing power or money. Procedural modelling is needed to solve these types of problems. Procedural modelling can be used to model individual buildings [2]. These building models can be used as models in procedurally generated cities [3].

Our method
Our method is based on general procedural modelling system, introduced by Muller et al [1], which is based on formal grammar by Chomsky [4]. Production rule in our method consists from a left and a right side. The left side of the production rule contains information about a non-terminal, which it applies to. The right side of the production rule contains operations. Our implementation allows user to use several operations. These can be categorized in three categories: scope changing functions, transformations and color/texture functions.

Example
Simple example, demonstrating available functions together with syntax used in our program.

Available functions
Simple application of the extrude function. We can perform either a pool extrude (left) or a box extrude (right)

Split production rule on multiple structures with different component split production rule.

Simple application of the component split production rule. We have selected only front walls of each structure in this image.

Example of transformations. There are four transformations available: translation, scaling, rotation and orbital rotation.

Results
1. “17. listopadu” dormitory
Our first model is based on the “17. listopadu” dormitory. We use low detail and low resolution textures for this model to show a difference between model with low quality textures and model with high quality textures.

Comparison of a model generated by our program with real building from a side angle. Please notice positioning of our model compared to real buildings, which is almost identical.

2. Prefab building on “Na Petřinách” street

Our second result presents a fully textured model. This model is also based on a real building located in Prague 6. We use textures with higher resolution to provide better visual output of our model.

Model of prefab building generated by our program

The two pictures above are both taken from a model generated by our program. They provides us with two detailed views of our model, showing the extent of implemented functionality. There are differences in comparison with the original building, however our model is a good overall approximation of the original building.

3. “Na Větrníku” dormitory
Our third example will display our model of “Větrník” dormitory. This model is also fully textured and combines various functionality. This model is our most complicated example. “Větrník” dormitory is a complex of three buildings - entrance hall and two blocks. However, our model consists of nine different objects. This was necessary due to the complicated ground plan of both blocks. Each block consists of four structures. These are positioned via transformations to form the final model, which looks like a solid model made out of one structure.

Conclusions & future work
Our focus in our bachelor thesis was to implement a program for procedural modelling of buildings. We described the method we implemented together with functions available to the user. We also briefly described the theory behind procedural modelling of buildings together with some other implementations. Then we tested our implementation by modelling buildings based on real buildings.

Further development of our program could include additional functions to either manipulate scope or change it, including simple roof generator, or allow us to generate non-rectangle pedestals for our buildings. Our existing functionality can also be improved, e.g. using relative paths in texture command.

References